May-June 1978
vol 4, no 3

creative computing

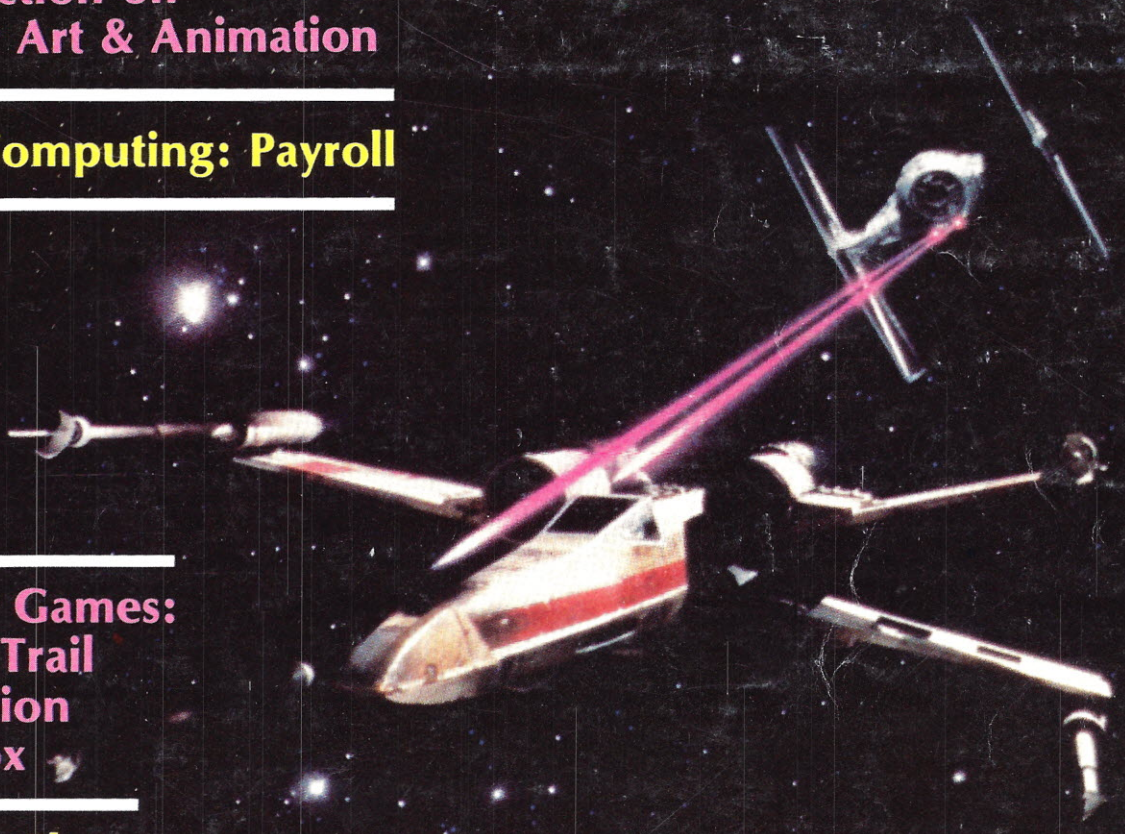
the #1 magazine of computer applications and software

Austria Sch 43.00
Belgium BF 94.00
Canada $2.50
Denmark Kr 15.00
Finland Mk 9.00
France F 12.00
Germany DM 6.00
Great Britain £ 1.30
Greece Dr 87.00
Holland Df 17.00
Italy L 2000
Norway Kr 13.00
Portugal Esc 73.00
Spain Ptas 160.00
Sweden Kr 11.00
Switzerland SF 6.00
USA $2.00

$2.00

Special Section on Computer Art & Animation

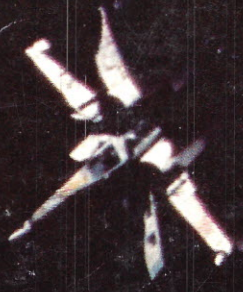
Business Computing: Payroll



Computer Games:
- Oregon Trail
- Art Auction
- Black Box

Computer Chess

The Perfect Puzzle?



Budget in BASIC

The Mechanical Mouse

Modelling the Cat Falling

Beginner's View of SAM76

Is Binary Search "Natural"?

READY for BUSINESS

We've got it all together—the cost effectiveness and reliability of our 6800 computer system with a high capacity 1.2 megabyte floppy disk system. . . PLUS—an outstanding new DOS and file management system.

1 MEGABYTE DISK SYSTEM

DMAF1 introduces a new level of capability to small computer systems. This disk system features two standard size floppy disk drives using the new double sided disk and two heads per drive. Usable storage space of over 600 kilobytes per drive, giving a total of over 1.0 megabyte of storage on line at all times. Ideal for small business applications, or for personal "super" systems.

DMA CONTROLLER

The controller occupies one main memory slot in an SS-50 bus and uses the Motorola MC-6844 DMA controller. The combination of a DMA type controller and double sided disks give the system speed of data transfer unobtainable with smaller drives.

OPERATING SYSTEM

To compliment this outstanding hardware we are supplying equally superior software. The disk operating system and file management system is called FLEX. It is one of the most flexible and complete DOS's available for small systems, but just as important; it is easy to use.

No one can match the variety of compatible peripherals offered by Southwest Technical Products for the SS-50 bus and the 6800 computer system. Now more than ever there is no reason to settle for less.

DMAF1 Disk System (assembled) $2,095.00

DMAF1 Disk System (kit). $2,000.00

68/2 Computer with 40K of memory (assembled) $1,195.00

SWTPC

SOUTHWEST TECHNICAL PRODUCTS CORPORATION
219 W. RHAPSODY
SAN ANTONIO, TEXAS 78216

CIRCLE 101 ON READER SERVICE CARD

PHOENIX, AZ
BYTE SHOP OF ARIZONA
24 WEST CAMELBACK (602) 263-0065

LITTLE ROCK, AR
COMPUTER PRODUCTS UNLTD.
2412 SO. BROADWAY (501) 666-2839

CARSON, CA
SUNSHINE COMPUTER CO.
20710 S. LEAPWOOD AVENUE (213) 327-9118

CITRUS HEIGHTS, CA
BYTE SHOP OF SACRAMENTO
6041 GREENBACK LANE (916) 726-2557

FOUNTAIN VALLEY, CA
ADVENTURES IN COMPUTING, INC.
8756 WARNER AVE. (714) 848-8388

LOOMIS, CA
KINGMONT ENTERPRISES, INC.
5900 KING ROAD (916) 652-8180

MT. VIEW, CA
BYTE SHOP OF MOUNTAIN VIEW, #1
1063 EL CAMINO REAL (415) 969-5464

SAN MATEO, CA
COMPUTER TERMINAL
309 SO. SAN MATEO DRIVE (415) 347-8894

SUNNYVALE, CA
RECREATIONAL COMPUTERS
1424 SOUTH MARY (408) 735-7480

VAN NUYS, CA
COMPUTER COMPONENTS
5848 SEPULVEDA BLVD. (408) 455-0640

FT. MYERS, FL
MICRO COMPUTER SYSTEMS INC.
13900 US 41 SOUTH (813) 481-3336

TAMPA, FL
MICRO COMPUTER SYSTEMS INC.
144 SOUTH DALE MABRY HIGHWAY (813) 879-4225

ATLANTA, GA
ATLANTA COMPUTER MART
5091-B BUFORD HWY (404) 455-0647

HAIKU, MAUI, HI
CAPACITY, INC.
P.O. BOX A (808) 575-2950

HONOLULU, HI
COMPACT COMPUTERS
P.O. BOX 10096 (808) 261-3281

BLOOMINGTON, IN
DATA DOMAIN
406 SO. COLLEGE AVENUE (812) 224-3607

FORT WAYNE, IN
DATA DOMAIN
2805 E. STATE BLVD. (219) 484-7511

PIKESVILLE, MD
MODULAR SYSTEMS, INC.
4005 SEVEN MILE LANE (301) 484-6322

ROCKVILLE, MD
COMPUTER WORKSHOP
1776 EAST JEFFERSON (301) 468-0463

TOWSON, MD
COMPUTERS, ETC.
13-A ALLEGHENY AVENUE (301) 296-0520

BURLINGTON, MA
THE COMPUTER STORE
120 CAMBRIDGE STREET (617) 272-8770

EDINA, MN
COMPUTER DEPOT
3515 W. 70TH STREET (612) 927-5601

ISELIN, NJ
COMPUTER MART OF NEW JERSEY
501 ROUTE #27 (201) 283-0600

EGGERTSVILLE, NY
CORSON COMPUTER CORPORATION
3834 MAIN STREET (716) 832-0662

HOLLIS, NY
SYNCHRO SOUND ENTERPRISES
193-25 JAMAICA AVENUE (212) 468-7067

CINCINNATI, OH
DATA DOMAIN
7694 CAMARGO ROAD (513) 561-6733

DAYTON, OH
DATA DOMAIN
1932 BROWN STREET (513) 223-2348

COLUMBIA, SC
BYTE SHOP OF COLUMBIA
2018 GREEN STREET (803) 771-7824

NASHVILLE, TN
SURYA CORPORATION
5755 NOLENSVILLE ROAD (615) 834-5638

CORPUS CHRISTI, TX
MICRO SYSTEMS SERVICES, INC.
5301 EVERHART, SPACE H (512) 855-4516

DALLAS, TX
COMPUSHOP
211 KEYSTONE PK, 13922 NORTH CENTRAL (214) 234-3412

HOUSTON, TX
COMPUTERLAND OF S. W. HOUSTON
6439 WESTHEIMER (713) 977-0909

HOUSTON, TX
ELECTROTEX
2300 RICHMOND AVENUE (713) 526-6934

HOUSTON, TX
THE MOS
1853 RICHMOND AVENUE (713) 527-8008

APPLETON, WI
SOUND WORLD
3015 W. WISCONSIN AVE. (414) 733-8539

RACINE, WI
COLORTRON TV
2111 LATHROP AVENUE (414) 637-2003

HUNTINGDON, CAMBS, ENGLAND
COMART LIMITED
39 GORDON ROAD, LITTLE PAXTON

MUNSTER, 4400 HILTRUP, WEST GERMANY
BASIS MICROCOMPUTER VERTRIEB
VON-FLOTOW-STRABE 5

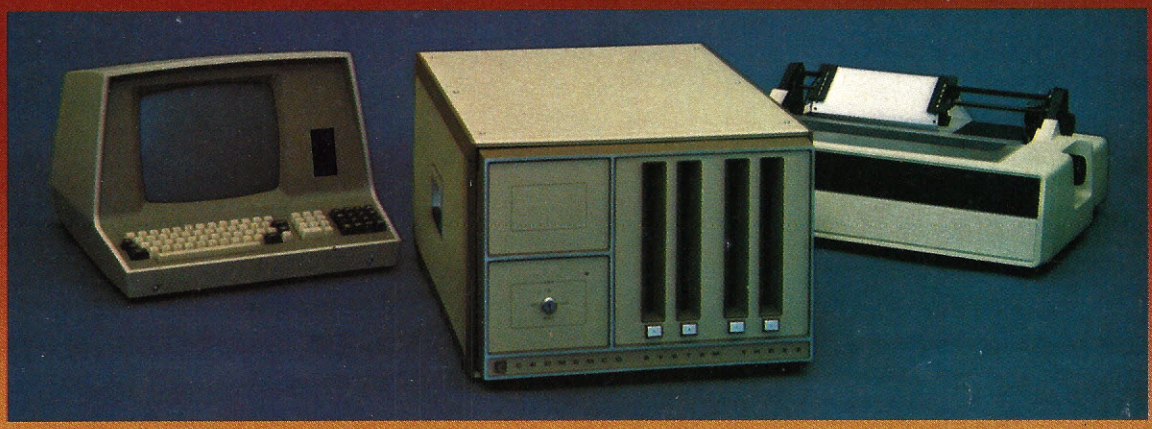
WEDEL 2000 (HOLSTEIN), WEST GERMANY
DIGITRONIC COMPUTERSYSTEME
BEI DER DOPPELEICHE 3-5

MEXICO 18, D.F. MEXICO
INTELEX, S. A.
P. A. DE LOS SANTOS 70

STOCKHOLM 29, SWEDEN
DATORISERING KONSULT AB
FACK 2, 10052

ZURICH, 8003 SWITZERLAND
COMICRO AG
BADENERSTRASSE 281

Now we can announce it– the multi-disk drive System Three Computer



A fast Z80 microcomputer with up to 512 kilobytes of RAM, 4 disk drives and 1 megabyte of disk storage—with CRT terminal and fast printer. Even an optional PROM programmer. Strong software support, too, like FORTRAN IV, Extended BASIC, and Macro Assembler.

PROFESSIONAL GRADE— FOR PROFESSIONALS

Chances are you've already heard that there is a Cromemco System Three Computer. We've proudly previewed it at WESCON on the West Coast and NYPC on the East Coast.

It's a complete system—processor, CRT terminal, line printer.

First it's fast—1 microsecond nominal execution time and 250 nanosecond cycle time.

Its equally fast RAM memory is large and enormously expandable—32 kilobytes expandable to 512 kilobytes. No danger of obsolescence from inadequate RAM capacity.

THE ONLY MICROCOMPUTER OFFERING 4 DISK DRIVES

Further, the System Three comes with two disk drives to give you 512 kilobytes of disk storage. Soft-sectored IBM format. Optionally, you can have four drives with 1 megabyte of storage.

There's disk protection, too, since in the LOCK position disks can't be ejected while they are running.

21-SLOT MOTHERBOARD

This new CS-3 is a computer that won't be outdated soon. It has a 21-card-slot slide-out motherboard and an S-100 bus so that you can plug in all sorts of support circuitry. The heavy-duty 30-amp power supply can easily handle all this.

BROAD S-100 SUPPORT

The S-100 is the bus that Cromemco so strongly supports with over a dozen plug-in circuits ranging from analog I/O to high-speed RAM memory with our bank-select feature.

TRULY POWERFUL SOFTWARE

You have to have software. And Cromemco is far in front there, too. Our FORTRAN IV, for example, is equal to the FORTRAN compilers on large mainframes. Further, it (and our other software) is low-priced.

Our 16K Z80 BASIC is one of the fastest and most capable. Full 14-digit precision.

There's also our Z80 Macro Assembler and Linking Loader. Uses Z80 mnemonics. Allows referencing FORTRAN common blocks.

SEE AT YOUR DEALER

You have to see the CS-3 to fully appreciate it and its low prices starting at $5990 in the rack mount version.

Better contact your dealer now.

Cromemco
incorporated
Specialists in computers and peripherals
2400 CHARLESTON RD., MOUNTAIN VIEW, CA 94043 • (415) 964-7400

CIRCLE 102 ON READER SERVICE CARD

Why Apple II is the world's best selling personal computer.

Which personal computer will be most enjoyable and rewarding for you? Since we delivered our first Apple® II in April, 1977, more people have chosen our computer than all other personal computers combined. Here are the reasons Apple has become such an overwhelming favorite.

Apple is a fully tested and assembled mainframe computer. You won't need to spend weeks and months in assembly. Just take an Apple home, plug it in, hook up your color TV* and any cassette tape deck—and the fun begins.

To ensure that the fun never stops, and to keep Apple working hard, we've spent the last year expanding the Apple system. There are new peripherals, new software, and a 16-chapter Owner's Manual on "How to Program in BASIC." There's even a free Apple magazine to keep owners on top of what's new.

Apple is so powerful and easy to use that you'll find dozens of applications. There are Apples in major universities, helping teach computer skills. There are Apples in the office, where they're being programmed to control inventories, chart stocks and balance the books. And there are Apples at home, where they can help manage the family budget, control your home's environment, teach arithmetic and foreign languages and, of course, enable you to create hundreds of sound and action video games.

When you buy an Apple II you're investing in the leading edge of technology. Apple was the first computer to come with BASIC in ROM, for example. And the first computer with up to 48K bytes RAM on one board, using advanced, high density 16K devices. We're working to keep Apple the most up-to-date personal computer money can buy. Apple II delivers the features you need to enjoy the real satisfaction a personal computer can bring, today and in the future.

15 colors & hi-resolution graphics, too.

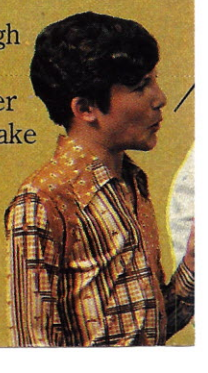
Don't settle for a black and white display! Connect your Apple to a color TV and BASIC gives you instant command of three display modes: Text, 40h x 48v Color-graphics in 15 colors, and a 280h x 192v High Resolution array that lets you plot graphs and compose 3-D images. Apple gives you the added capability of combining text and graphics, too.

Back to basics, and assembly language too.

Apple speaks three languages: fast integer BASIC, floating point BASIC for scientific and financial applications, and 6502 assembly language. That's maximum programming flexibility. And, to preserve user's space, both integer BASIC and monitor are permanently stored in 8K bytes of ROM, so you have an easy to use, universal language instantly available. BASIC gives you graphic commands: COLOR=, VLIN, HLIN, PLOT and SCRN. And direct memory access, with PEEK, POKE and CALL commands.

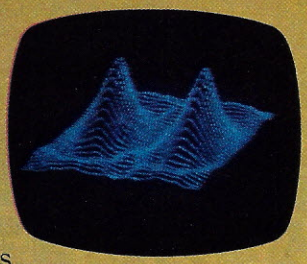
Software: Ours and yours.

There's a growing selection of pre-programmed software from the Apple Software Bank—Basic Finance, Checkbook, High Resolution Graphics and more. Now there's a User Section in our bank, to make it easy for you to obtain programs developed



by other Apple owners. Our Software Bank is your link to Apple owners all over the world.

Alive with the sound of music.



Apple's exclusive built-in speaker delivers the added dimension of sound to your programs. Sound to compose electronic music. Sound to liven up games and educational programs. Sound, so that any program can "talk" back to you. That's an example of Apple's "people compatible" design. Another is its light, durable injection molded case, so you can take Apple with you. And the professional quality, typewriter-style keyboard has n-key rollover, for fast, error-free operator interaction.

Apple is the proven computer.

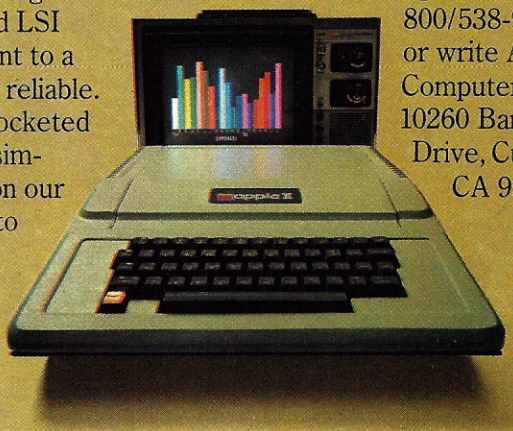
Apple is a state-of-the-art single board computer, with advanced LSI design to keep component count to a minimum. That makes it more reliable. If glitches do occur, the fully socketed board and built-in diagnostics simplify troubleshooting. In fact, on our assembly line, we use Apples to test new Apples.

Apple peripherals are smart peripherals.

Watch the far right column of this ad each month for the latest in our growing family of peripherals. We call them "intelligent interfaces." They're smart peripherals, so you can plug them in and run them from BASIC without having to develop custom software. No other personal computer comes close to Apple's expandability. In addition to the built-in video interface, cassette I/O, two A/D game paddles, and two more A/D inputs, Apple has eight peripheral slots, three TTL inputs and four TTL outputs. Plus a powerful, state-of-the-art switching power supply that can drive all your Apple peripherals, including two disks.

Available now.

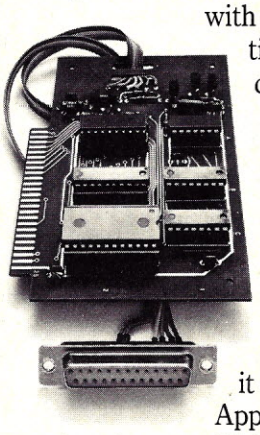
Apple is in stock and ready for delivery at a store near you. Call us for the dealer nearest you. Or, for more details and a copy of our "Consumer Guide to Personal Computers," call 800/538-9696 or write Apple Computer, Inc., 10260 Bandley Drive, Cupertino, CA 95014.



New from Apple.

Introducing the Apple Communication Interface

Apples of the world unite! Now you can, with our new intelligent communication interface card. Just plug it in and it turns your Apple into an intelligent terminal that can go on line with other terminals, time-sharing computers and, especially, with other Apples. You can even play Tele-Pong! Everything you need is on one small card. With a modem, it enables your Apple to communicate by phone at 110/300 baud RS232 full duplex I/O. The card is fully assembled and tested and has all required software in on board ROM. It's controlled by simple BASIC commands. And it's available from stock.



Peripherals in stock

Hobby Board, Parallel Printer Interface, Communication Interface.

Coming soon

High speed serial printer interface, General purpose serial interface, Printer II, Printer IIA, Disk II, Monitor II.

* Apple II plugs into any standard TV using an inexpensive modulator (not included).

apple computer®

CIRCLE 103 ON READER SERVICE CARD

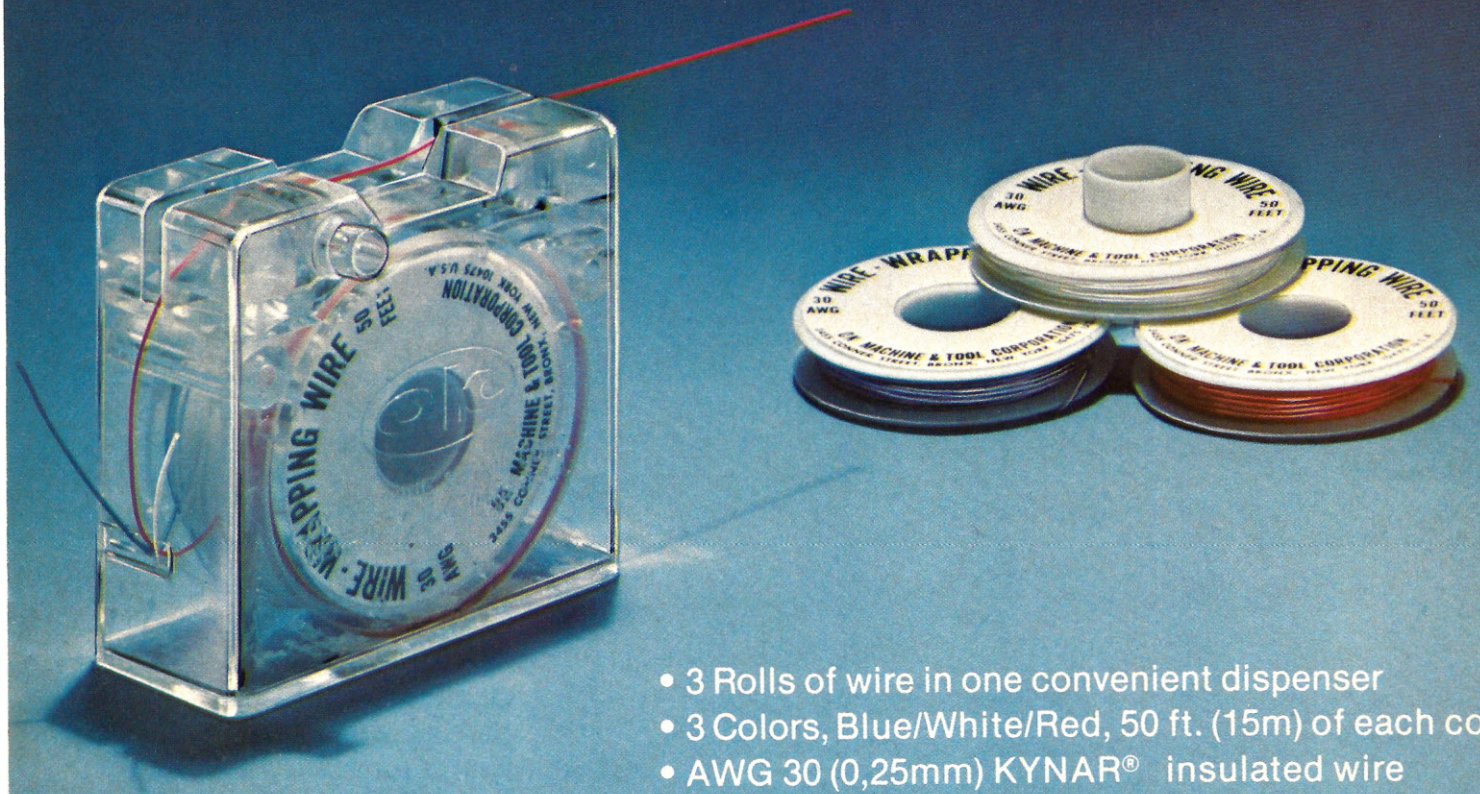
Programming is a snap! I'm halfway through Apple's BASIC manual and already I've programmed my own Star Wars game.

Those math programs I wrote last week–I just rewrote them using Apple's mini-assembler and got them to run a hundred times faster.

Apple's smart peripherals make expansion easy. Just plug 'em in and they're ready to run. I've already added two disks, a printer and the communications card.

TRI-COLOR

WIRE CUTTING AND STRIPPING DISPENSER



- 3 Rolls of wire in one convenient dispenser
- 3 Colors, Blue/White/Red, 50 ft. (15m) of each color
- AWG 30 (0,25mm) KYNAR® insulated wire
- Built-in cutting plunger cuts wire to desired length
- Built-in stripper strips 1″ of insulation
- Easily refillable
- For wire-wrapping and other applications.

Model No. WD-30-TRI
Patent Pending

WD-30-TRI	DISPENSER WITH WIRE	$5.95
R-30-TRI	TRI-COLOR REPLACEMENT SPOOLS	$3.95

Kynar® Pennwalt

MINIMUM BILLING $25.00 / ADD SHIPPING CHARGE $1.00 / NEW YORK STATE RESIDENT ADD APPLICABLE TAX

OK® MACHINE & TOOL CORPORATION 3455 CONNER ST. BRONX, N.Y. 10475 (212) 994-6600/Telex 125091

CIRCLE 117 ON READER SERVICE CARD

in this issue ...

articles

SPECIAL SECTION:
COMPUTER ART & ANIMATION

BUSINESS COMPUTING: PAYROLL

fiction & foolishness

things to do - games

reviews & resources

departments

May-June 1978 Volume 4, Number 3 Consecutive Issue No. 21

Creative Computing magazine is published bi-monthly by Creative Computing, P.O. Box 789-M, Morristown, NJ 07960. (Editorial office: 51 Dumont Place, Morristown, NJ 07960. Phone: (201) 540-0445.)

Domestic Subscriptions: Institutional 1-year $15, 3-year $40; individual 1-year $8, 2-year $15, 3-year $21. Subscription orders, change of address, P.O. Form 3579 to Creative Computing, P.O. Box 789-M, Morristown, NJ 07960. Call 800-631-8112 toll-free to order a subscription (to be charged only to a bank card).

Foreign Subscriptions:

Great Britain: Institutional 1-year £9.40, 2-year £11.90, 3-year £25.00; Individual 1-year £6.25, 3-year £16.90. Orders to Vincent Coen, 313 Kingston Road, Ilford, Essex, IGI-IPJ, England.

Europe: 1-year subscription. Austria Sch 180, Belgium BF 4000, Denmark Kr65, Finland Mr40, France Fr50, Germany DM25, Greece Dr360, Holland Dfl27, Italy L8500, Norway Kr55, Portugal Esc320, Spain Ptas 700. Sweden Kr45, Switzerland SF25. Orders to Pan Atlantic Computer Systems GmbH, Frankfurter Strasse 78, D61 Darmstadt, German Fed Rep.

Australia: R. J. Hoess, Electronic Concepts Pty. Ltd., 52-58 Clarence St., Sydney NSW 2000, Australia.

Other Countries: 1-year $12, 2-year $23, 3-year $33 (surface postage, U.S. dollars). Orders to Creative Computing, P.O. Box 789-M, Morristown, NJ 07960.

Second class postage paid at Morristown, New Jersey and at additional mailing offices.

Publisher
David H. Ahl

Editor-in-Chief
Stephen B. Gray

Managing Editor
Burchenal Green

Associate Editor
Steve North

Higher Education Editors
Merl Miller
A. Kent Morton

Technology Editor
Alan Salisbury

Careers Editor
Eleanor Corr

Humor Editor
Jack Ludwig

Art Director
Elyse J. Fox

Marketing Manager
Linda Harrison

Bookkeeper
Jeanne Tick

Retail Marketing
Linda Eckerstrom

Customer Service
Ethel Fisher

Subscriptions
Nancy Hammond
Carol Cassata

Book Service
Barbara Shupe
Katherine McKenzie

Order Processing
Thomas Bass

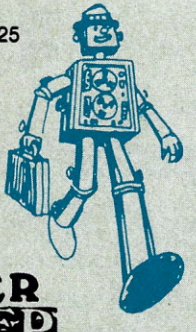
Advertising Sales

Western States, Texas
Jules E. Thompson
Hearst Building, Suite IIII
5 Third Street
San Francisco, CA 94103
(415) 362-8547

Southern California
Bert Charlton
2560 Via Tejon
Palos Verdes Estates, CA 90274
(213) 378-8361

Mid-Atlantic, Northeast
Charles Lynch
36 Sohier Street
Cohasset, MA 02025
(617) 383-6136

Elsewhere
Burchenal Green
(201) 540-0445



...notices...

Inventory Systems and Computers

The fourth annual lecture series sponsored by The Department of Mathematical Sciences of The Johns Hopkins University and The Johns Hopkins Press, will be held in Baltimore **June 12-16, 1978.** The subject will be the analysis of inventory systems on computers. Included will be lectures, on-line computer demonstrations, and computer workshops.

The principal speaker will be Professor Eliezer Naddor, who will present 10 Inventory Computer Modules covering such topics as Price-breaks, Probabilistic Demands and Leadtimes, Simulation, Optimal and Heuristic Decisions, and Information Storage and Retrieval in Multi-Item Systems. His lectures will be accompanied by on-line demonstrations of all modules. Participants will be able to use the modules during the daily workshops.

The registration fee is $300. It includes 10 documented and illustrated computer programs, use of the university, five luncheons, and two dinners.

Persons interested in presenting papers at the conference and/or giving on-line demonstrations on inventory systems and computers are invited to write to:

1978 Lecture Series Committee Department of Mathematical Sciences, The Johns Hopkins University, Baltimore, MD 21218.

Our Face is Red

In the article, "Structured Software for Personal Computing" in the March-April 1978 issue (p 58), a line is missing at the top of the center column on page 60: "manageable set of relatively simple;" and Figures 9 and 10 on page 63 somehow got interchanged.

* * *

Also, in the Jan-Feb 1978 *Catalogue,* the price of the Integral Data Systems impact printer is given (p 16) as $475. We should have said $745; the latest price is $799.

Camp Retupmoc

Six one-week programs in computer science will be offered this summer on the campus of Rose-Hulman Institute of Technology, Terre Haute, Indiana.

The program, called Camp Retupmoc, is an intensive investigation into the world of computers with the people who make them do useful things. Featured are lectures on BASIC programming along with talks from computer leaders in business and industry. Sessions on microprocessors and careers in computing are included.

Full cost, including tuition, room, and board, is $135. Starting dates are from **June 11** to **July 23.** For applications and more information contact Dr. John Kinney, Director, Camp Retupmoc, Rose-Hulman Institute of Technology, Terre Haute, Indiana, 47803.

Conference on Computers in Undergraduate Curricula

The Ninth Conference on Computers in Undergraduate Curricula, CCUC/9, is scheduled for **June 12-14, 1978,** at the University of Denver in Colorado. The primary purpose of CCUC/9 is to "promote effective use of computers in undergraduate education," and papers have been solicited from "all institutions of higher education with concern for undergraduate instruction. All disciplines are eligible for inclusion in the conference."

In contrast to the other conferences in the CCUC series, papers dealing with computer uses in computer science courses, with user service aspects of computer centers, and with the use of programmable calculators in undergraduate education "will be considered."

Chairman: William S. Dorn, Dept. of Mathematics, University of Denver, Denver, CO 80208.

The Cover

If you don't recognize our cover photo, then you're one of the few computer addicts who hasn't seen the immensely popular "Star Wars" movie. For information on how computers were used to create some of the spectacular scenes, see the article "The Digital Brush," an interview with animator Larry Cuba, in this issue. And get ready for the second of what may become a series: Star Wars II.

PLETE SATISFACTION IS OUR GUARANTEE YOUR COMPLETE SATISFACTION IS OUR GUARANTEE YOUR COMPLETE SATISFACTION

Huge Discounts!

HOBBY WORLD ELECTRONICS

Savings up to 70% on major brand IC parts and computer kits. For complete IC listings write for our catalog.

KITS

WAMECO	S-100 P.C. Boards	
	8K RAM	28.00
	8080 CPU	28.00
	12-Slot Mother Board	33.00
ITHACA AUDIO	S-100 P.C. Boards	
	8K RAM	28.00
	Z-80 CPU	28.00
SOLID STATE MUSIC	S-100 Kits & Bare Boards	
MB-3 2K/4K EPROM	Uses 1702A EPROMS	
	Kit w/o EPROMS	59.95
MB-4 4K STATIC RAM	Kit	95.00
	Bare Board	25.95
MB6A 8K STATIC RAM	Kit	129.95
	Bare Board	25.95
MB7 16K STATIC RAM	Kit	435.00
	Bare Board	25.95
MB8 8K/16K EPROM	Uses 2708's	
	Kit Less EPROMs	75.95

SPECIAL

Cybercom, div. of Solid State Music

S-100 MUSIC SYNTHESIZER KIT $145.00

KITS

SOLID STATE MUSIC		
MB9 STATIC PROM/RAM	Kit Less Memory	72.00
VB1A VIDEO INTERFACE	Kit	129.95
	Bare Board	25.95
IO2 PARRELL I/O	And Kludge Kit	49.95
	Bare Board	25.95
IO4 2+2 I/O	Kit	139.95
SYNTHESIZER SB-1 MUSIC	Kit with Software	145.00
MT-1	15-Slot Mother Board	39.95
XB-1 EXTENDER BOARD	Bare Board	8.99
SSM 8080 MONITOR V1	ON 2-2708	47.00
	ON 8-1702A	47.00

BOOKS

The BASIC Workbook	$5.50
Programming Proverbs	6.95
Discovering BASIC	6.85
COBOL with Style	6.95
Advanced BASIC	7.95
Basic BASIC	8.95
Standard Dictionary of Computers & Information Processing	16.95
Game Playing with Computers	16.95
Game Playing with BASIC	6.95
Introduction to BASIC	8.95
Home Computers: 210 Questions and Answers Volume 2: Software	6.95
Minicomputers	13.95
Microcomputer Dictionary and Guide	19.95
Microprocessors	10.95
Microprocessor Basics	10.95
Modern Data Communications	8.95
Home Computers: 210 Questions and Answers Volume 1: Hardware	7.95
Understanding Integrated Circuits	4.95

BOOKS

Semiconductor Circuit Elements	6.95
Digital Experiments	8.95
Digital Signal Analysis	19.95
Digital Troubleshooting	9.95
110 CMOS Digital IC Projects	5.95
Fundamentals and Applications of Digital Logic Circuits	7.95
400 Ideas for Design, Volume 2	13.75
Analysis and Design of Digital Circuits and Computer Systems	22.50
Finite State Fantasies	1.25
Telephone Accessories You Can Build	4.95
Basic Electronic Switching for Telephone Systems	6.95
Basic Carrier Telephony	6.85
How to Get the Most Out of Your Low-Cost Electronic Calculator	4.95
Calculator Users Guide and Dictionary	12.95

74L$00 TTL

74L$00	.21	74L$138	.70
74L$02	.21	74L$139	.70
74L$04	.24	74L$151	.65
74L$08	.21	74L$153	.66
74L$10	.21	74L$154	1.00
74L$14	.85	74L$157	.62
74L$20	.23	74L$160	.82
74L$21	.23	74L$161	.82
74L$22	.23	74L$162	.82
74L$30	.23	74L$163	.82
74L$32	.30	74L$164	.98
74L$37	.31	74L$174	.75
74L$38	.31	74L$175	.79
74L$42	.60	74L$190	.90
74L$47	.75	74L$191	.90
74L$48	.72	74L$192	.90
74L$73	.35	74L$196	.80
74L$74	.35	74L$197	.80
74L$75	.53	74L$221	1.06
74L$76	.37	74L$257	.71
74L$86	.36	74L$258	.70
74L$90	.52	74L$266	.26
74L$92	.52	74L$283	.72
74L$93	.52	74L$365	.55
74L$109	.36	74L$366	.55
74L$112	.36	74L$367	.55
74L$113	.36	74L$368	.55
74L$114	.36	74L$386	.39
74L$125	.46	81L$95	.77
74L$126	.46	81L$96	.77
74L$132	.75	81L$97	.77
		81L$98	.77

74XX

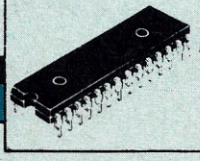
7400	.11	7482	.50
7401	.13	7483	.62
7402	.13	7485	.75
7403	.13	7486	.26
7404	.15	7489	1.75
7405	.13	7490	.40
7406	.16	7491	.51
7407	.23	7492	.40
7408	.17	7493	.40
7409	.17	7494	.60
7410	.13	7495	.60
7411	.18	7496	.60
7412	.13		
7413	.25	74107	.28
7414	.61	74109	.22
7416	.24	74121	.29
7417	.22	74122	.38
7420	.13	74123	.48
7423	.25	74132	.65
7425	.25	74141	.70
7426	.22	74145	.65
7427	.17	74150	.88
7430	.13	74151	.61
7432	.23	74153	.61
7437	.21	74154	.95
7438	.21	74157	.55
7439	.25	74161	.55
7440	.13	74163	.55
7441	.70	74164	.85
7442	.37	74165	.90
7443	.59	74170	1.69
7444	.59	74173	1.10
7445	.65	74174	.85
7446	.62	74175	.75
7447	.59	74176	.69
7448	.60	74177	.70
7450	.13	74180	.65
7451	.13	74181	1.75
7453	.13	74190	.95
7454	.13	74191	.95
7460	.14	74192	.79
7470	.26	74193	.80
7472	.21	74195	.49
7473	.21	74221	.86
7474	.27	74251	1.00
7475	.45	74365	.62
7476	.28	74366	.62
7480	.31	74367	.62
7481	.95	74368	.62

SPECIAL

2708 FULL SPEC. 450NS 8 FOR $90.00

SUPPORT DEVICES

6820	8.00
6850	8.00
8212	3.45
8214	8.00
8216	3.75
8224	3.50
8228	6.25
8226	3.85
8238	7.95



MEMORY

1702A	3.75
2708	12.50
2716	22.50
21L02 450ns	1.25
21L02 250ns	1.60
1101	.50
2114	8.50

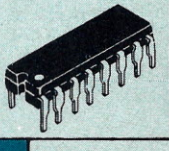
MICROPROCESSOR

8080A	11.50
Z-80	24.95
Z-80A	34.95
6800	16.50

4001	.16
4002	.16
4006	.99
4007	.16
4008	.79
4009	.37
4010	.37
4011	.16
4012	.16
4013	.29
4014	.74
4015	.74
4016	.29
4017	.79
4018	.79
4019	.23
4020	.84
4021	.89
4022	.89
4023	.16
4024	.67

CMOS

4025	.16
4028	.74
4030	.22
4035	.98
4040	.99
4049	.34
4050	.34
4051	.89
4052	1.15
4053	.89
4066	.59
4071	.19
4081	.23
4082	.23
4507	.39
4510	1.05



5716 W. Manchester Ave.
Suite #5
Los Angeles, CA 90045

TELEPHONE ORDERS:
Call (213) 641-4200

master charge THE INTERBANK CARD

BANKAMERICARD VISA

☐ Cash
☐ COD

Charge My
☐ M.C.
☐ BAC (VISA)

Exp. Date:

* — —

Sig.

Name

Address

City State Zip

☐ Send your complete catalog, quickly.
☐ Please send me the following items I have listed below:

Qty.	Stock No.	Price
	Postage/Handling	$1.50

Satisfaction 100% Guaranteed

California Residents Add 6% Sales Tax

Note: Minimum Order $10.00, 5% Discount over $100.00 on I.C.'s only.

NEW 1978 IC MASTER

- Over 40,000 IC's listed.
- Over 2,000 pages.
- Updated every 90 days.

Retail Value
$55.00
Your Price
$46.00

IS OUR GUARANTEE YOUR COMPLETE SATISFACTION IS OUR GUARANTEE YOUR COMPLETE SATISFACTION IS OUR GUARANTEE

CIRCLE 109 ON READER SERVICE CARD

IS OUR GUARANTEE YOUR COMPLETE SATISFACTION IS OUR GUARANTEE YOUR COMPLETE SATISFACTION IS OUR

al... editorial... editor

Things to Come

Over the next five years I expect dramatic changes in every aspect of the small computer field. Those of us now in the field will be overwhelmed, at least in numbers, by people today who have never heard of a personal computer. This is somewhat akin to the situation in 1920 when radio amateurs, who had for years been a growing but close-knit group, all of a sudden, with the advent of commercial AM radio, found themselves in a minority of radio users. Companies that had been catering to hams switched over to production of commercial radios as a new consumer industry leaped into life. Oh sure, some manufacturers stuck with the hams and over the years there were new entrants, but the real growth was in commercial radio.

Today, the TRS-80, PET, VideoBrain, and Atari Video Computer System are the first of what promises to be a broad, expanding line of commercial personal computers. More and more, the video game systems will have keyboard and memory options, and new computers will be announced at the Toy Fair or Consumer Electronics Show rather than at computer industry shows. How often have you seen Atari or Coleco at a personal computing show or the NCC? Yet it is from companies like these that I expect major future developments. (This is one reason that at *Creative Computing* we cover these "other" shows and product profiles of video games and the like).

A parallel development to the completely assembled, neatly packaged commercial computer system will be systems dedicated to a single function or group of functions. For example, no longer will you buy *one* general-purpose computer, but you will buy one for text editing, one for library cataloging, one for games, one for music synthesis, one for CAI, and so on. As prices come down to $300 and lower it just won't make sense to buy the peripherals to do all these functions on one system, when several dedicated, individual systems can be bought for the same or less cost.

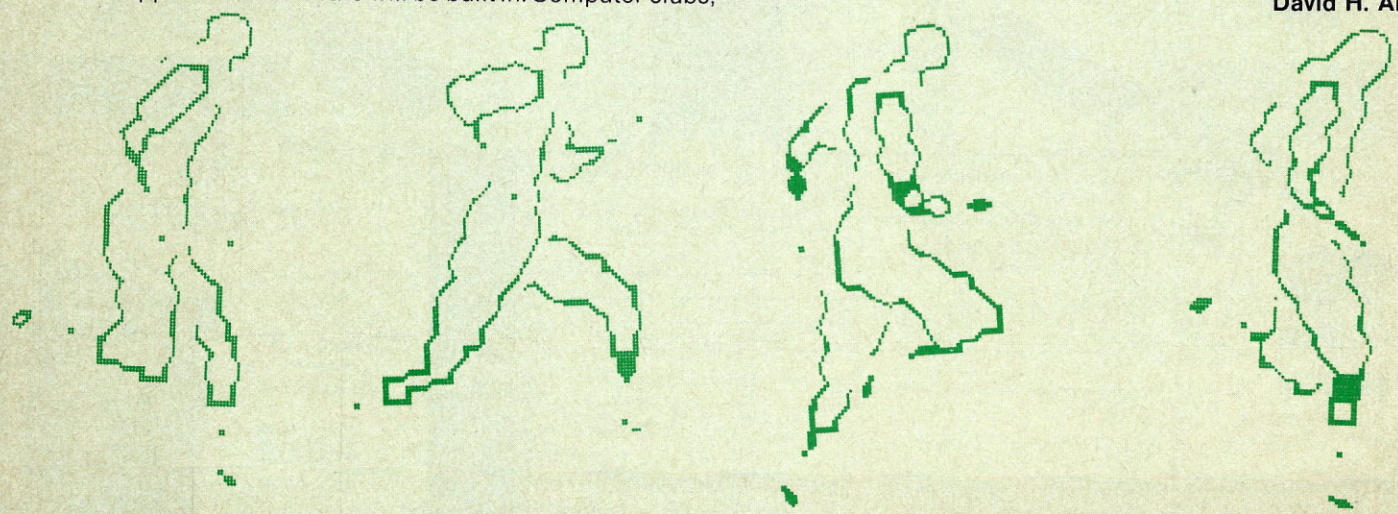
The user, of course, will not have to learn to program in Basic or other computer language since all the systems and applications software will be built in. Computer clubs, therefore will lose one of their primary functions — software interchange. Indeed, the typical buyer of a commercial personal computer, like buyers of AM radios in the 1920's, will have little interest in a computer club anyway. After all, they're buying their computer for one or more specific purposes, not for the fun of building it, or writing software, or any of the other reasons that most people have bought their own computers for the past three years.

Another parallel development that will profoundly influence the use of small computers will be the establishment of one or more low-cost digital communication networks. The recently-announced Bell Data Network (ACS) may be overkill for home users but no matter what the form, home and small-business users will have access to high-speed data communications. Not only will users have access to data bases containing all types of encyclopedic data, stock-market data, and the like, but also the small business will be able to receive orders from field sales representatives, acknowledge orders, quote prices, and perform all the other data-communication functions now available only to larger businesses with their own data nets.

In forecasting all this, I don't mean to imply that the current cult of personal computer users will die out. Quite the contrary, they will continue to exist just as radio amateurs did. Some will gravitate toward packaged commercial systems while others will continue in computing as a hobby. There will be side-by-side development between hobbyists and packaged systems users, some overlap and much synergy. All in all, the future of small computing will continue to be intellectually challenging and exhilarating, it will expand at an increasing pace, and in ten years most people will regard a personal computer as commonplace as a transistor radio or pocket calculator today.

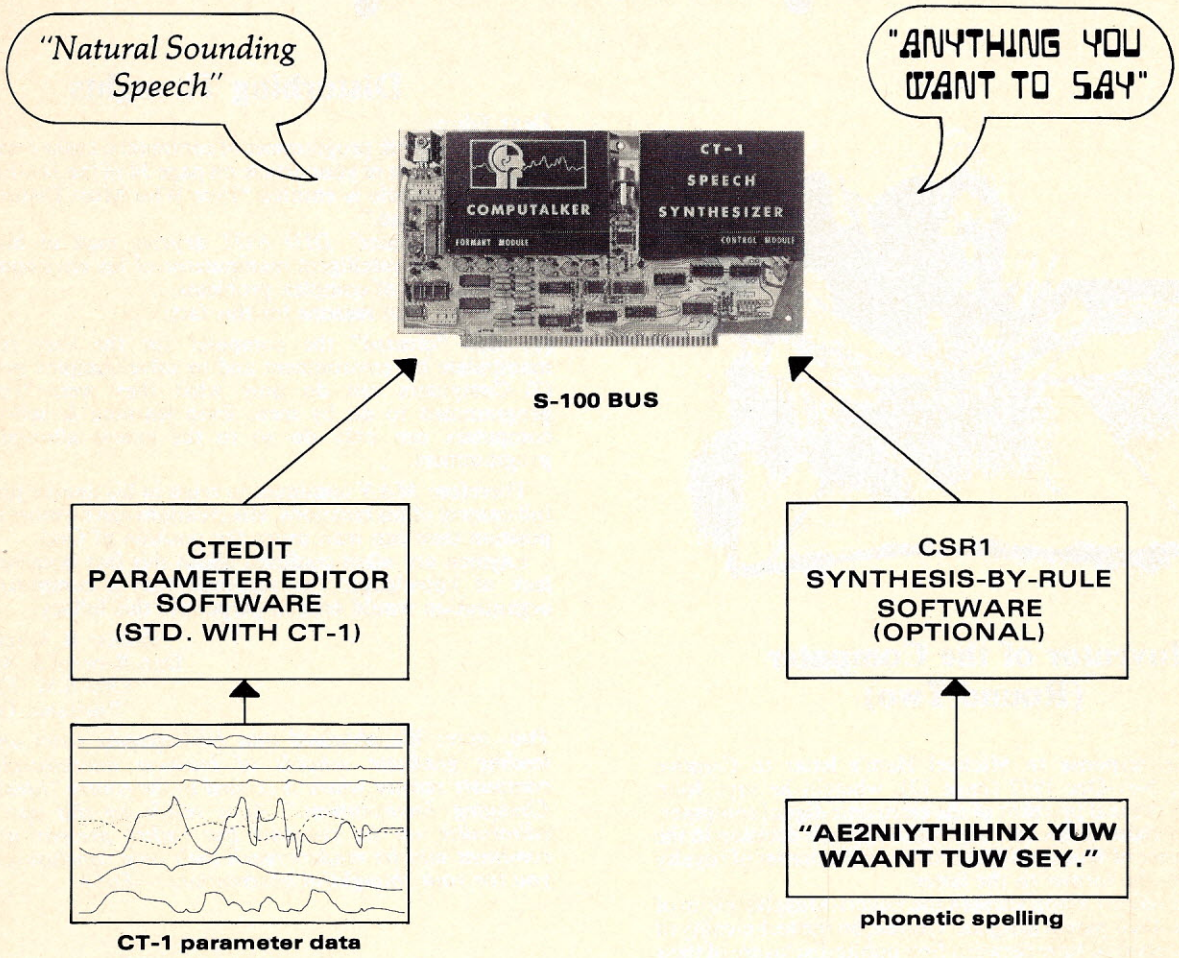
We at *Creative Computing* intend to be there too, growing and changing with the field. We hope you'll be with us.

David H. Ahl



Lillian Schwartz. *Olympiad*, 1971: Stills from computer-animated film. Courtesy Bell Laboratories, Murray Hill, N.J.

YOU CAN HAVE BOTH . . .



ONLY COMPUTALKER GIVES YOU THE CHOICE

No other speech synthesizer available offers you the flexibility of the Computalker Model CT-1. The parameter editor program, CTEDIT, allows direct manipulation of all 9 control parameters. CTEDIT and parameter data tapes are furnished with the CT-1 synthesizer board.

The optional CSR1 software package translates ASCII phonetic text strings to speech output. CSR1 is simple to use and is the easiest way to create new speech. CSR1 can also be called as a subroutine from user's code for applications involving program controlled voice output.

The CSR1 phonetic rule system generates control parameters in the same form as used by CTEDIT. Thus, it is possible to further edit the output of the rule system, to achieve natural sounding speech output with minimum effort.

CT-1 assembled & calibrated	**395.00**
CSR1 phonetic rules software	**35.00**
Calif. residents add 6% sales tax	

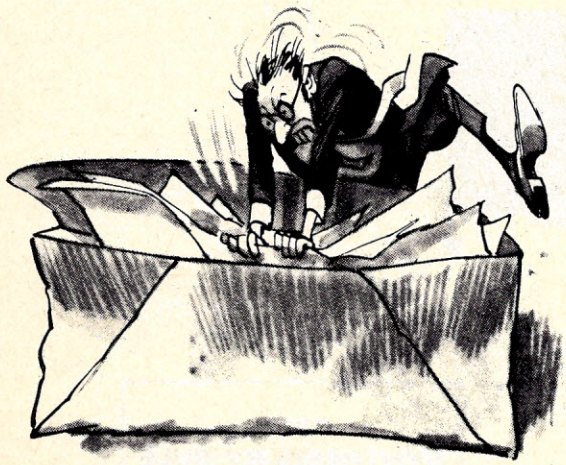
Software is available on CPM 8", North Star 5¼", CUTS, TARBELL, MITS ACR, Paper Tape.

COMPUTALKER CONSULTANTS

1730 21st Street, Suite AE, Santa Monica, CA 90404 (213) 392-5230/828-6546

CIRCLE 119 ON READER SERVICE CARD

put...input/output...in



Inventor of the Computer (Round Two)

Dear Editor:

This is in response to Michael Ham's letter to *Creative Computing,* Nov-Dec 1977 (page 12), wherein he says John Atanasoff is the real inventor of the electronic digital computer. He bases his claim solely on Judge Earl Larson's decision in the 1971 court case of Honeywell vs Univac on the matter of royalty payments by the former to the latter.

Actually, Judge Larson vacated the Eckert-Mauchly claim of royalties due their patent assignee, Univac, on the technicality of an excess interval of time between first public disclosure of their patentable product and the date of formal application for their landmark patent. Having thus overridden the 25-year standing of the Eckert-Mauchly patent, and having ignored four prior court tests (including IBM's) of this same patent in which the Univac claim had been legally upheld and royalties awarded to Univac, Judge Larson then usurped powers vested in the Bureau of Patents and unilaterally bestowed fatherhood of the computer on Prof. Atanasoff.

During that trial in Minneapolis in the Spring and Summer of 1971, Honeywell introduced over 25,000 documents in evidence, and Univac added almost 8,000 other trial exhibits. Besides the letter Mr. Ham mentioned concerning Mauchly's visit to Atanasoff to see the latter's "computer," other letters produced in court showed clearly that Atanasoff was unable (on the occasion of Mauchly's visit) to demonstrate the "Device" to Mauchly after trying for several days. Hence that device can in no way be called a computer (a device which computes). Another exhibit presented during the trial was a letter from Atanasoff to Mauchly congratulating the latter for having succeeded where he (Atanasoff) had failed.

Judge Larson also neglected to account in his monumental decision for John Atanasoff's failure, during the 28 years since the original Eckert-Mauchly computer's successful operation at the University of Pennsylvania in 1943, to make any public claim of prior invention of the computer until the issue was raised by Honeywell during this trial in Minneapolis. Since Atanasoff was not entitled to any royalties because he had not patented this idea, his only benefit from Judge Larson's unprecendented decision was a big ego trip, undeserved as it is.

Ernest J. Tursich
818 Forest Ave.
Northfield, MN 55057

Disturbing Thoughts

Dear Editor:

As an owner & programmer of computers, I take exception to the fifth portion of your article on page 34 of the Jan-Feb, 1978, issue. This article is entitled "Five Who Most Disturbed The Thought of Man."

The "Computer" *DID NOT* deprive man of his unique position as an intelligent manipulator of his environment and creative solver of complex problems.

There are two reasons for this fact:

(1) Man "created" the computer for the very reason to manipulate his environment and to solve complex problems.
(2) Computers can do only what they were "creatively" programmed to do by men. Even learning & self-initiating computers can only do so to the extent allowed by the programmers.

Therefore *MAN* continues to reign in his unique position in full control of his creations. His creations can no more usurp his position than can man usurp the position of God.

Laymen are often confused about this fact because of their lack of knowledge. I'm surprised that a computer related organization would seek to perpetrate this fallacy.

Robert B. Kircher, P.E.
Dan Rowley & Associates
1300 S.W. Fifth Ave.
Portland, OR 97201

Pub. note: We obtained that bibliography from one of the leading graduate schools of business administration and computer science where it is used in the course "Ideas and the Changing Environment." As a result we did not want to editorialize on its contents but rather present it without comment and let readers reach their own conclusions. Thank you for your thoughtful observations.—DHA

The Last Bug

Dear Editor:

The poem "The Last Bug" published in the Nov-Dec 1977 *Creative Computing,* p 131, appears to be an *n*th generation modification of the original poem written by Lou Ellen Davis, wife of HP-65 Users Club member Perry Davis. The poem is "the Perfect Program" and first appeared in *Computers and Automation,* Aug 1967, p 43. Somehow it has acquired the "Author Unknown" label causing people to feel free to modify it, resulting in the one you published.

A. John Martellaro, Jr.
HP-65 Users Club
2929 Los Amigos Ct., Apt B
Las Cruces, NM 88001

What's Your Favorite?

Dear Editor:

Concerning future projects for my company, what ANSI-level programming languages and/or operating system software would you and your readers like to see available in the $40-$50 per copy price range over the next two years? If you would print this letter, perhaps your readers would send their ideas to me.

Michael Clark, President
Cybermate
R.D. #3—Box 192A
Nazareth, PA 18064

The Small Businessman's System

MSI now has a low-cost business computer system which is designed to help small business the same way big computers help big business...by saving time and money.

Regardless of the size or type of your business...service...manufacturing... professional...or sales...the MSI 6800 Business System will significantly and profitably improve the speed, accuracy, and efficiency of your business operations.

A System For Every Application

The system starts with the MSI 6800 computer, a 64K processor which is fast, simple, reliable, and will hold up to 56K of RAM. Next, interface the computer with the popular MSI FD-8 Floppy Disk Memory System, or the soon to be released MSI Quad Density Double Sided Floppy Disk Memory System. Or for those really big small business jobs, MSI will soon have a fixed/removable hard disk drive which will give you 76 megabytes of memory storage.

Add a video terminal and a hard copy printer and you will have at your command the worlds finest microcomputer business system.

Small Computers for Big Jobs

MSI has a lot more information about computer systems for small business. Send for our free Microcomputer Products Catalog and get all the details about our many fine products.

Midwest Scientific Instruments

220 W. Cedar
Olathe, Kansas 66061
(913) 764-3273
TWX 910 749 6403 (MSI OLAT)
TELEX 42525 (MSI A OLAT)

CIRCLE 108 ON READER SERVICE CARD

COMPLEAT COMPUTER CATALOGUE

We welcome entries from readers for the "Compleat Computer Catalogue" on any item related, even distantly, to computers. Please include the name of the item, a brief evaluative description, price, and complete source data. If it is an item you obtained over one year ago, please check with the source to make sure it is still available at the quoted price.

Send contributions to "The Compleat Computer Catalogue," *Creative Computing,* P.O. Box 789-M, Morristown, NJ 07960.

BOOKS AND BOOKLETS

ACADEMIC COMPUTING DIRECTORY

A new directory published by the Human Resources Research Organization will make it easier for this nation's schools and colleges to use computers for teaching and learning by introducing them to *other* schools and colleges that are using computers successfully. The *Academic Computing Directory* provides information on how computers are being used for instructional purposes by 367 educational institutions ranging in level from elementary schools to major universities. The Directory identifies the "exemplary" institutions and provides information on the brands of computers they have, the purposes for which these computers are used, and the major reason(s) the institutions were selected as "exemplars" of academic computing. $3.95.

HumRRO, 300 N. Washington St., Alexandria, VA 22314.

CIRCLE 180 ON READER SERVICE CARD

VENDOR LITERATURE

TDL CATALOG

The latest catalog from Technical Design Labs covers a dozen hardware items from boards such as the ZPU card, the Z16 memory module and the Video Display Board, to complete systems such as the five computers in the Xitan Alpha series; and software that includes 8K BASIC, Super-BASIC, the Zapple monitor, text editor, macro-assembler, text output processor, Fortran IV, and a data-base management system. All are described in detail in this 16-page catalog.

Technical Design Labs, Inc., Research Park, Bldg. H, 1101 State Road, Princeton, NJ 08540. (609) 921-0321.

CIRCLE 181 ON READER SERVICE CARD

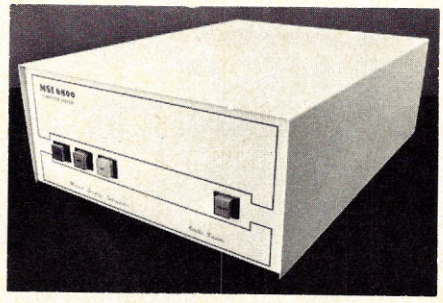
NORTH STAR CATALOG

North Star Computers, manufacturer of the HORIZON computer and S-100-bus peripheral products, has just produced a new 16-page color product catalog. The catalog provides detailed information on all products and software available from North Star. The catalog is offered to computer hobbyists, business users, dealers and computer clubs free of charge. It is available through computer stores or directly from:

North Star Computers, Inc. 2547 Ninth Street, Berkeley, CA 94710. (415) 549-0858.

CIRCLE 182 ON READER SERVICE CARD

COMPUTERS

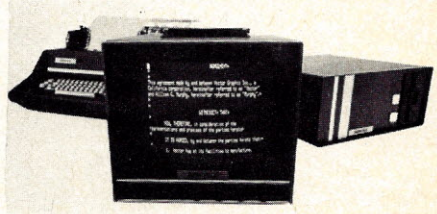


NEW MSI 6800 COMPUTER

The MSI 6800 computer system, suitable for use in business, industrial, or educational environments, employs the popular SS-50 bus architecture, and individual modules of the MSI system may be used to upgrade an existing SWTP 6800 system if desired. The motherboard contains 16 positions for full-sized system boards. Front-panel push-buttons for power, reset, IRQ, and NMI are provided. The MSI CPU Board contains sockets for 4K of EPROM memory, 128 bytes of RAM, in addition to a restart vector PROM. A 14411 baud rate generator as well as a 6875 clock generator are included on the CPU Board. This permits the system clock to be run at 2 MHz, separate from the baud rate generator if desired. $595 kit, $895 wired.

Midwest Scientific Instruments, 220 W. Cedar, Olathe, KS 66061. 913/764-3273.

CIRCLE 183 ON READER SERVICE CARD

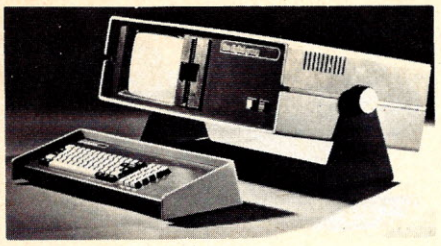


VECTOR GRAPHIC WORD PROCESSING SYSTEM

A low-cost, video-based text-editing system has been introduced by Vector Graphic. Called Memorite, the three-piece system comprises the firm's Vector 1+ microcomputer with floppy-disk drive, a high-resolution Hitachi 12-inch monitor, and a Diablo HyTerm printer. The printer offers a speed of 540 words per minute and interchangeable printwheels. Memorite permits full editing with capabilities for revision, addition or deletion, insertions, margin adjustment, text block transfer and finished text display in the CRT prior to printing. Additional features include variable line and page length; character, line and page spacing; and right justification. The disk basic system included with Memorite will accommodate software for other applications such as legal billing, accounting and inventory control. $7,950.

Vector Graphic Inc., 790 Hampshire Road, A+B, Westlake Village, CA 91361. (805) 497-6853.

CIRCLE 184 ON READER SERVICE CARD

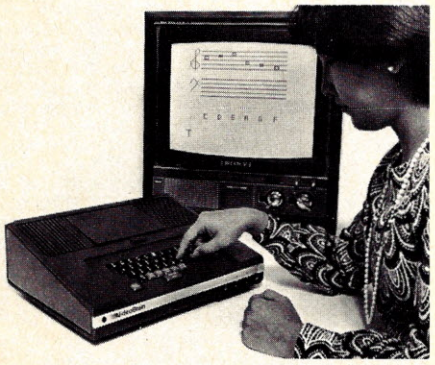


BYTEMASTER COMPUTER

The Digital Group's first completely integrated computer package, the Bytemaster will be available for shipment May 1, 1978. The Bytemaster features either 18K or 32K memory, but will support up to 64K memory if desired. Designed with expansion in mind, the Bytemaster is fully wired to support various external peripherals. You may add a printer, monitor, and additional digital cassette, mini-disk, or standard disk drives by simply plugging into any of the four available input/output ports. Dressed in a professional metal cabinet mounted on a heavy duty metal yoke, the top-of-the-line Master 4 model (mini-disk, 32K, assembled) is $3,245.

W. V. Honeyman, The Digital Group, Inc., P.O. Box 6528, Denver, CO 80206. (303) 777-7133.

CIRCLE 185 ON READER SERVICE CARD



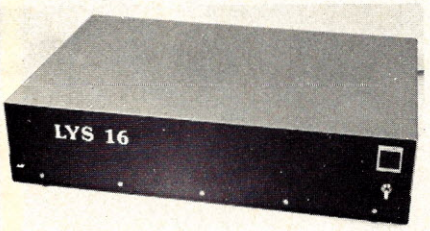
VIDEOBRAIN COMPUTER

Said to be the first home computer with a preprogrammed library of educational, home management and entertainment programs, the VideoBrain will be sold through department stores and specialty electronic stores. The computer comes with everything necessary to hook it up to a TV and start running programs. The AC adapter, TV hookup cord, antenna switch box, two joy sticks, and three introductory cartridge programs are included. The VideoBrain console has 36 input keys and weighs 10 pounds. Input to the computer is made either by using the keyboard as you would use a typewriter (there are 71 distinguishable input symbols) up to four X/Y joysticks, and preprogrammed cartridges with a wide range of useful and fun programs.

The VideoBrain has expandability built in. The present model has jacks for expansion to tape cassettes, printer and telephone. Built into the VideoBrain is the basic text and timekeeping programs. The text program allows the user to type and edit a message of 7 lines and 16 characters per line. He can change the color of the screen or the size of the letters and he can store the message for retrieval later on. ROM cartridges are available for such programs as Finance, Cash Management, Real Estate Analysis, Stock Valuation, Music Teacher, Math Tutor, and a variety of games, from blackjack to pinball. VideoBrain has a suggested retail price of $500.

Umtech, 150 South Wolfe Rd., Sunnyvale, CA 94086. (408) 737-2680.

CIRCLE 186 ON READER SERVICE CARD



LYS-16 COMPUTER

This 16-bit microcomputer was developed by members of Lysator Computer Club at the University of Linkoping, Sweden. The CPU is National Semiconductor's IMP-16, with 61 machine instructions. There are also four general registers on a 16-word hardware LIFO-stack. To all this has been added an advanced interrupt system which makes it possible to connect up to 64 peripheral units on four different levels. The computer is connected to a standard TV set through the TTY-compatible terminal, TERMILVS. This terminal has 64 ASCII characters and will display 25 rows of 64 characters on the TV. It has interesting features such as a graphical mode and word highlighting. Secondary storage is an audio cassette recorder, but a floppy-disk unit will soon be released. Software consists of a conversational assembler and editor, a BASIC with the rather unique range of ±1.4E-9864 to ±8.8E9862, and since LYS-16, to a large extent, is compatible with the IMP-16 system, one can use most of the software developed by National. Prices are not definite for foreign customers yet, but further information can be obtained through:

AB ATEW, P.O. Box 125, S-692 00 Flen, Sweden.

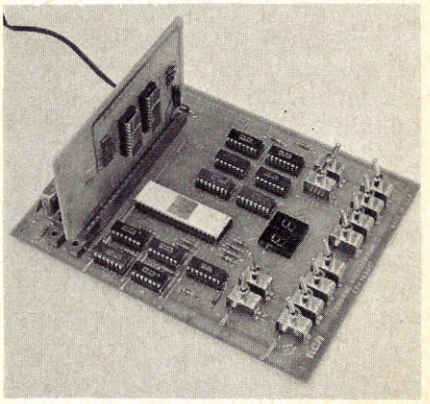
CIRCLE 187 ON READER SERVICE CARD

XYCON III COMPUTER

Computer Systems Unlimited's Xycon III is an all-in-one system consisting of a 24x80 high-resolution CRT and controller with character intensification, blinking, underscoring, and reverse video—in any combination, all on a character-by-character basis. Using a 63-key keyboard with 16-key numeric and 8-key alternate-action pad, the system supports high-resolution graphics (256x256) or special (APL) and scientific or foreign alphabets. The 32K RAM memory board is expandable in 16K increments to 65K bytes, and usable as 8-bit or 16-bit word memory. Dual floppy disks are built in, with an intelligent controller that uses its own firmware to do formatting, etc. The CPU board uses an 8085A MPU. The Xycon III can support 16 users in time-sharing mode, by adding multi-user software and memory to total 48K to 64K. The Standard Xycon III is $9,220.

Computer Systems Unlimited, P.O. Box 870, Milpitas, CA 95035. (408) 262-6271.

CIRCLE 188 ON READER SERVICE CARD



RCA COSMAC MICROTUTOR II

Intended especially for engineers, students, and hobbyists who wish to understand and use microprocessors, RCA Solid State's COSMAC Microtutor II, CDP18S012, is a complete basic microcomputer system available for quick and easy hands-on operating and programming experience. The new RCA COSMAC Microtutor II, preassembled and containing its own regulated power supply, is based on the RCA CDP1802 CMOS 8-bit microprocessor and supersedes the original Microtutor CDP18S011. The new CDP18S012 provides input via eight binary toggle switches and output on two seven-segment LED hexadecimal digit displays plus a Q LED output. Additional toggle switches are provided for all the required controls to examine and alter memory locations and to initiate program execution. Microtutor II is provided with 256 bytes of CMOS RAM on a memory card which attaches to the base through a standard 44-pin connector. Microtutor II has a crystal clock for stabilized timing applications and a memory protect switch which inhibits the memory write operation to prevent an improperly running program from writing into itself. $195.

For further information and copies of the Product Description PD9: RCA Solid State Division, Box 3200, Somerville, NJ 08876, or from RCA Solid State distributors.

CIRCLE 189 ON READER SERVICE CARD

it's a good bet the company you bought your computer from doesn't even make peripherals!

It's no great surprise! Most computer companies got their start in the digital logic end of the business. They were great at building calculators and later computers but when it came right down to it, most just didn't have the experience necessary to build the peripherals to support their computer products. And that left a vacuum!

At Heath we had the advantage. Our years of experience in electronic kit design gave us plenty of background with not only digital logic but mechanical and video design as well. And our assembly manuals and documentation are world-famous for easy to understand instructions.

We built the world's first digital color television, a unique fully synthesized FM tuner, digital frequency counters, clocks – even a digital bathroom scale.

So when we entered the personal computing market we had the "know-how" to build not only our outstanding H8 and H11, 8 and 16-bit computers, but, in addition, a complete line of supporting peripheral kits!

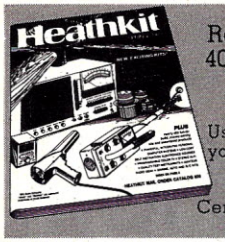
Select the H9 Video Terminal, the H10 Papertape Reader/Punch, and very soon our own, complete, Floppy Disk system. Each was designed with the systems approach in mind. Each was conceived to integrally mesh with not only our own computers, but through our set of sophisticated interfaces, most others as well. And in that way we're making every effort to fill the vacuum the others left!

So when you're ready to communicate with your computer turn to Heath. We've got the peripheral kits you'll need and at prices you can afford.

Maybe the company who sold you your computer didn't think about peripherals – but we sure did! And come to think about it maybe that's why you should come to Heath...in the first place.

Heathkit Computers

FREE Heathkit Catalog



Read about nearly 400 money-saving, fun-to-build electronic kits.

Use coupon to send for your mail order catalog or bring coupon to a Heathkit Electronic Center for your catalog.

HEATH
Schlumberger

Heath Company, Dept. 355-410
Benton Harbor, MI 49022

Please send me my FREE Catalog. I am not on your mailing list.

Name_____

Address_____

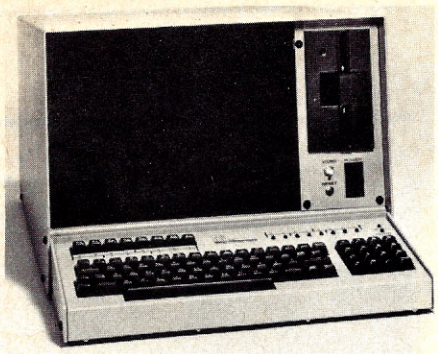
City_____ State_____

CP-146 Zip_____

AVAILABLE LOCALLY AT HEATHKIT ELECTRONIC CENTERS
(Units of Schlumberger Products Corporation) Retail prices on some products may be slightly higher.

ARIZONA: Phoenix, 85017, 2727 W. Indian School Rd., Phone: 602-279-6247; **CALIFORNIA: Anaheim, 92805,** 330 E. Ball Rd., Phone: 714-776-9420; **El Cerrito, 94530,** 6000 Potrero Ave., Phone: 415-236-8870; **Los Angeles, 90007,** 2309 S. Flower St., Phone: 213-749-0261; **Pomona, 91767,** 1555 Orange Grove Ave. N., Phone: 714-623-3543; **Redwood City, 94063,** 2001 Middlefield Rd., Phone: 415-365-8155; **Sacramento, 95825,** 1860 Fulton Ave., Phone: 916-486-1575; **San Diego (La Mesa, 92041),** 8363 Center Dr., Phone: 714-461-0110; **San Jose (Campbell, 95008),** 2350 S. Bascom Ave., Phone: 408-377-8920; **Woodland Hills, 91364,** 22504 Ventura Blvd., Phone: 213-883-0531; **COLORADO: Denver, 80212,** 5940 W. 38th Ave., Phone: 303-422-3408; **CONNECTICUT: Hartford (Avon, 06001),** 395 W. Main St. (Rte. 44), Phone: 203-678-0323; **FLORIDA: Miami (Hialeah, 33012),** 4705 W. 16th Ave., Phone: 305-823-2280; **Tampa, 33614,** 4019 West Hillsborough Ave., Phone: 813-886-2541; **GEORGIA: Atlanta, 30342,** 5285 Roswell Rd., Phone: 404-252-4341; **ILLINOIS: Chicago, 60645,** 3462-66 W. Devon Ave., Phone: 312-583-3920; **Chicago (Downers Grove, 60515),** 224 Ogden Ave., Phone: 312-852-1304; **INDIANA: Indianapolis, 46220,** 2112 E. 62nd St., Phone: 317-257-4321; **KANSAS: Kansas City (Mission, 66202),** 5960 Lamar Ave., Phone: 913-362-4486; **KENTUCKY: Louisville, 40243,** 12401 Shelbyville Rd., Phone: 502-245-7811; **LOUISIANA: New Orleans (Kenner, 70062),** 1900 Veterans Memorial Hwy., Phone: 504-722-6321; **MARYLAND: Baltimore, 21234,** 1713 E. Joppa Rd., Phone: 301-661-4446; **Rockville, 20852,** 5542 Nicholson Lane, Phone: 301-881-5420; **MASSACHUSETTS: Boston (Peabody, 01960),** 242 Andover St., Phone: 617-531-9330; **Boston (Wellesley, 02181),** 165 Worcester Ave. (Rt. 9 just west of Rt. 128), Phone: 617-237-1510; **MICHIGAN: Detroit, 48219,** 18645 W. Eight Mile Rd., Phone: 313-535-6480; **E. Detroit, 48021,** 18149 E. Eight Mile Rd., Phone: 313-772-0416; **MINNESOTA: Minneapolis (Hopkins, 55343),** 101 Shady Oak Rd., Phone: 612-938-6371; **MISSOURI: St. Louis (Bridgeton), 63044,** 3794 McKelvey Rd., Phone: 314-291-1850; **NEBRASKA: Omaha, 68134,** 9207 Maple St., Phone: 402-391-2071; **NEW JERSEY: Fair Lawn, 07410,** 35-07 Broadway (Rte. 4), Phone: 201-791-6935; **Ocean, 07712,** 1013 State Hwy. 35, Phone: 201-775-1231; **NEW YORK: Buffalo (Amherst, 14226),** 3476 Sheridan Dr., Phone: 716-835-3090; **Jericho, Long Island, 11753,** 15 Jericho Turnpike, Phone: 516-334-8181; **Rochester, 14623,** 937 Jefferson Rd., Phone: 716-244-5470; **White Plains (North White Plains, 10603),** 7 Reservoir Rd., Phone: 914-761-7690; **OHIO: Cincinnati (Woodlawn, 45215),** 10133 Springfield Pike, Phone: 513-771-8850; **Cleveland, 44129,** 5444 Pearl Rd., Phone: 216-886-2590; **Columbus, 43229,** 2500 Morse Rd., Phone: 614-475-7200; **Toledo, 43615,** 48 S. Byrne Rd., Phone: 419-537-1887; **PENNSYLVANIA: Philadelphia, 19149,** 6318 Roosevelt Blvd., Phone: 215-288-0180; **Frazer (Chester Co.), 19355,** 630 Lancaster Pike (Rt. 30), Phone: 215-647-5555; **Pittsburgh, 15235,** 3482 Wm. Penn Hwy., Phone: 412-824-3564; **RHODE ISLAND: Providence (Warwick, 02886),** 558 Greenwich Ave., Phone: 401-738-5150; **TEXAS: Dallas, 75201,** 2715 Ross Ave., Phone: 214-826-4053; **Houston, 77027,** 3705 Westhelmer, Phone: 713-623-2090; **VIRGINIA: Alexandria, 22303,** 6201 Richmond Hwy., Phone: 703-765-5515; **Norfolk (Virginia Beach, 23455),** 1055 Independence Blvd., Phone: 804-460-0997; **WASHINGTON: Seattle, 98121,** 2221 Third Ave., Phone: 206-682-2172; **WISCONSIN: Milwaukee, 53216,** 5215 W. Fond du Lac, Phone: 414-873-8250.

CIRCLE 105 ON READER SERVICE CARD



TEI COMPUTER/TERMINAL

TEI, Inc., has another model in their new processor terminal series, the MCS-PT112/32, a complete and self-contained computer system with display, disk storage, a full keyboard and a 12-slot motherboard. It may be used either as a standalone processor or as a processor terminal in a larger system. The MCS-PT112/32 includes a 15-inch high-resolution monitor with a face plate of smokey plexiglass to reduce glare and enhance type visibility, a full upper- and lower-case ASCII keyboard with eight user-designated special function keys and a 16-key numeric cluster pad. One Shugart SA-400 mini-floppy disk drive is standard. The 12-slot mainframe contains a CPU board with an 8080 processor. 32K static RAM memory is provided with additional RAM as an optional item. A disk controller which will handle three mini-drives. The minidrive media is soft-sectored and has a capacity of 90 KB unformatted (80.6 formatted). The video controller board uses a 24 X 80 format with many special features. Software provided with the system includes CP/M operating system and SuperBASIC, a 20K interpreter. $4,795.

CMC Marketing Corp., 5601 Bintliff, Suite 515, Houston, TX 77036. (713) 783-8880.

CIRCLE 190 ON READER SERVICE CARD

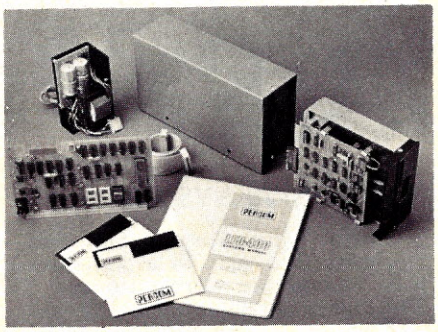
MICROCOMPUTER WITH DOUBLE-DENSITY FLOPPIES

Digital Systems has introduced a microcomputer system with dual-drive, double-density floppy disks for less than $5,000. Designated the Micro-2, this compact system is housed in a single cabinet with two Shugart floppy-disk drives. The single computer board features a Z-80 CPU, 32K or 64K RAM, four RS232 serial interfaces, and a real-time clock. The disk controller can use either IBM 3740 format or a double-density format of 571K bytes per diskette (77 tracks of 58 sectors with 128 bytes per sector). With optional double-sided drives, the system can store up to 2.3 Megabytes. The Micro-2 comes complete with both the comprehensive CP/M disk operating system and complete hardware diagnostics. Extensive accounting software is available. Other software, including CBASIC, BASIC-E and FORTRAN, is also available. Complete system with two single-sided drives: 32K, $4,995; 64K, $6,090. With two double-sided drives: 32K, $5,695; 64K, $6,795. With four single-sided drives: 32K, $7,040; 64K, $8,135.

Digital Systems, 6017 Margarido Drive, Oakland, CA 94618. (415) 428-0950.

CIRCLE 191 ON READER SERVICE CARD

PERIPHERALS



MINIFLOPPY FOR S-50 BUS

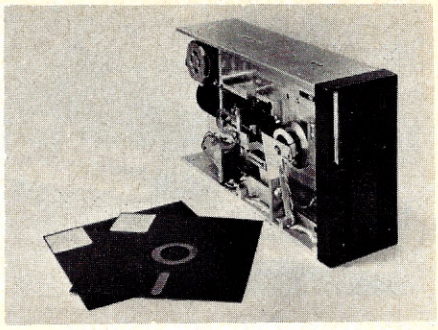
PerCom Data's LFD-400 is a minifloppy disk memory system for the S-50 bus. A complete one-drive LFD-400 system includes a controller PC board, PROMware disk operating system, disk drive and drive power supply, interconnecting cable, two minidiskettes, an operator's manual, and a compact enclosure to house the drive and drive power supply.

The controller board, which is installed in an S-50 bus slot of the host computer, includes special low-voltage-drop regulators, a proprietary "bit shifting" compensation circuit, an inactivity time-out circuit to increase drive motor life, and provision for 3K bytes of PROM. The LFD-400 PROMware DOS, miniDOS, allows S-50 bus owners to use their existing software with simple patches.

The miniDOS program includes load and save routines, and permits "crash-proof" data storage and retrieval since the disk may be protected. MiniDOS is contained in a 2708 EPROM, and is ready on power-up. The LFD-400 uses the Shugart SA-400 drive.The LFD-400 with one drive is $599.95; with two drives, $999.95; with three, $1399.95.

PerCom Data Company, Inc., 318 Barnes, Garland, TX 75042. (214) 276-1968.

CIRCLE 192 ON READER SERVICE CARD

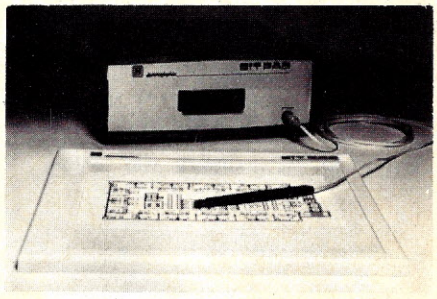


FOUR-HEADED VOICE-COIL FLOPPY

The industry's first "four-headed" flexible disk drive, which will store up to 3.2 megabytes of data in the space required by a standard-size floppy drive, is the new PerSci Model 299 Diskette Drive, interfacing to 8080, 6800 and Z-80 based systems as well as minicomputers. The Model 299 is a dual-headed, dual-diskette drive reading and writing both sides of two 8-inch diskettes. Data can be encoded in single or double density in IBM-compatible soft-sectored formats or expanded hard- and soft-sectored formats on IBM Diskette I, II, IID or equivalent media. The drive will store up to 1 megabyte of data in IBM type format, 1.6 megabytes unformatted single density and up to 3.2 megabytes in unformatted double density encoding. $1,595.

PerSci, Inc., 12210 Nebraska Ave., West Los Angeles, CA 90025. (213) 820-3764.

CIRCLE 193 ON READER SERVICE CARD



BIT PAD

Summagraphics Corporation announces "an innovative first in computer input devices," the Bit Pad, "more than a digitizer, a flexible input peripheral," and designed specifically for small system users. Bit Pad is a full-capability, digitizer permitting ease of entry of positional information. The Bit Pad is designed for fast, low-cost data collection of X, Y values. The small size (11" x 11") and compact design make the Bit Pad completely portable and adaptable to a wide variety of applications. Bit Pad is easily interfaced to any microcomputer currently on the market as it is equipped with a byte-oriented 8-bit parallel output. Applications exist in medical treatment, opinion sampling, education, real estate, design, games, research, computer animation and a limitless variety of additional uses. $555.

Summagraphics Corp., 35 Brentwood Ave., Fairfield, CT 06430. (203) 384-1344.

CIRCLE 194 ON READER SERVICE CARD

DISK SYSTEM UPGRADES HEATHKIT H8 TO Z80

INFO 2000 Corporation has a disk system for Heathkit H8 computers. Now Heathkit H8 users may add the INFO 2000 Disk System and simultaneously upgrade their 8080 computer to a Z80 system by replacing the Heathkit 8080 CPU board with the INFO 2000 Z80/Disk Adapter Board. The complete disk system includes PerSci dual diskette drives, power supply, case, intelligent controller, adapter, cables and disk monitor in EPROM. The adapter board contains the Z80 microprocessor and all support chips, 7K of EPROM, 1K of scratchpad RAM for the disk monitor, and all necessary logic for interfacing the disk system to the Heathkit H8.

The H8 can now operate in either of two switch-selectable modes. One mode enables continued use of the H8 EPROM monitor with the existing Benton Harbor software. No modification is required, and the H8 will now perform at Z80 CPU speed and use the extended Z80 instruction set. The second mode supports the INFO 2000 disk monitor, and other software adapted by INFO 2000 for use with all their disk systems, including the TDL software library and CP/M. $2,750.

INFO 2000 Corp., 20630 S. Leapwood Avenue, Carson, CA 90746. (213) 532-1702.

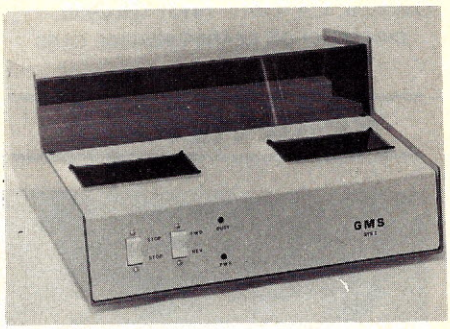
CIRCLE 195 ON READER SERVICE CARD

IBM SELECTRIC PRINTER FOR MICRO OUTPUT

The Micro Computer Devices SELECTERM is a fully converted IBM Selectric II Typewriter whose conversion to a printer enables immediate use with any microcomputer. The SELECTERM may be connected directly to either a parallel or serial port, with all inputs at standard TTL level. No additional software is required, since all logic is in an internal PROM. The SELECTERM includes a special typing element that produces all ASCII and full upper and lower case alphanumeric characters. Also included are table command, backspace, vertical tab and bell. All necessary electronics and cable sets are provided along with documentation for unpacking, connection, testing, theory of operation, and schematics. Special features may be ordered including dual pitch, correcting feature, pin-feed platen in a choice of 13 sizes, and a noise-reduction feature. The SELECTERM can be used as a typewriter since none of the typing capabilities have been affected by the conversion to a printer. The SELECTERM may be purchased only through dealers. $1,650.

Contact your computer store, or write Micro Computer Devices, 960 E. Orangethorpe, Bldg. F, Anaheim, CA 92801. (714) 992-2270.

CIRCLE 196 ON READER SERVICE CARD

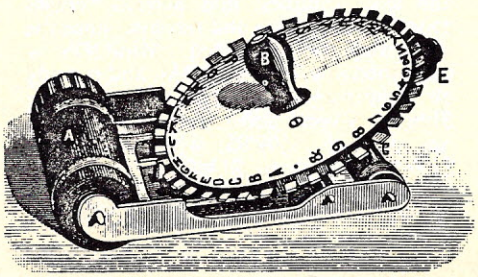


TAPE-DRIVE SYSTEM

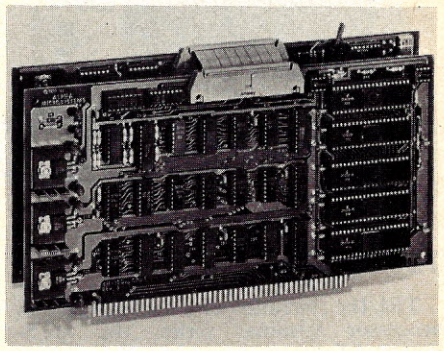
General Micro-Systems' new SYS I tape-drive subsystem, is a high-capacity mass storage for micro-computers. The SYS I records bi-phase Manchester code at 1600 bits per inch on ANSI-specified data cassettes with a transfer rate of 2000 characters per second at 10 IPS. The tape record (block) is variable length, which gives the highest efficiency of storage space on tape, "unlike the 128 or 256 byte fixed length records, where all bytes must be recorded whether used or not.)) A 10-byte record may be followed by a 32-Kbyte record. The user program may dynamically load the next record, operating as a batch data processing system, with an unlimited amount of data. Over 700-Kbytes may be recorded on one side of a cassette using large records. Rewind time is less than 30 seconds at over 120 IPS. One to four drives may be connected to the computer through the interface board. Single drive, $595; dual drive, $969; S-100 interface board, $168.

Bob Smith, General Micro-Systems, 12369 West Alabama Place, Lakewood, CO 80228.

CIRCLE 197 ON READER SERVICE CARD



MISC. HARDWARE

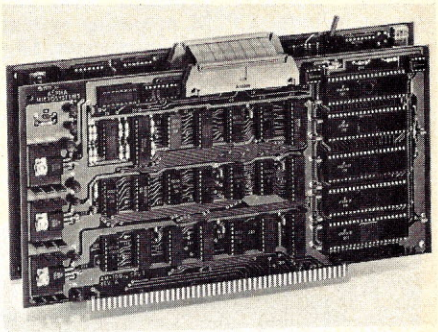


EPROM PROGRAMMER

Smoke Signal Broadcasting has a new low-cost 2708 EPROM programmer. The POP-1 interfaces to the conpany's P-38-1 and P-38-FF EPROM boards, which are SS-50 bus-compatible products. Complete software is provided on audio cassette. An adaptive programming technique allows most 2708's to be programmed in 15 seconds instead of the usual one and a half minutes. A separate self-contained power supply is used for the programming voltage, insuring sufficient current capability to program EPROM's from any manufacturer. $149.

Smoke Signal Broadcasting, P.O. Box 2017, Hollywood, CA 90028. (213) 462-5652.

CIRCLE 198 ON READER SERVICE CARD

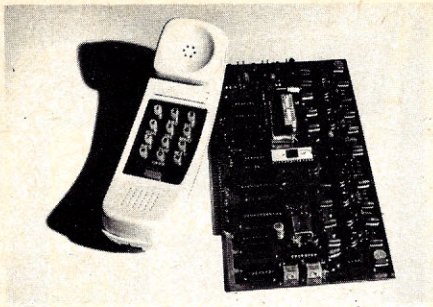


S-100 16-BIT MICRO

The AM-100, an S-100-bus-compatible 16-bit microprocessor board set, replacing 8-bit processors, offers multi-tasking, multi-user timesharing in an advanced disk operating system environment. Utilizing Western Digital's WD-16 microprocessor, the AM-100 provides 16-bit flexibility and speed with 11-digit floating-point arithmetic and an on-board realtime clock. Many S-100-bus peripherals are supported including static memory, memory paging and I/O facilities. The AM-100 provides a multi-pass Macro-Assembler, ALPHA-BASIC compiler, ALPHALISP, SORT, ISAM and various other utilities.

John French, Alpha Microsystems, 17875 Sky Park North, Suite N, Irvine, CA 92714. (714) 957-1404.

CIRCLE 199 ON READER SERVICE CARD

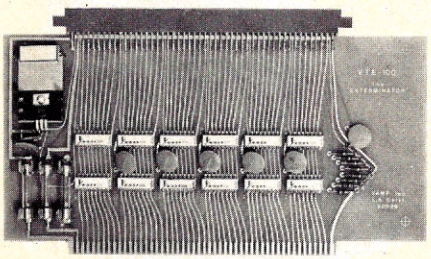


TOUCH-TONE INTERFACE

For those interested in bringing the microcomputer into the home, MK Enterprises has a Dual Tone Multi-Frequency (DTMF) transceiver board that interfaces your S-100 micro computer to the Touch-Tone telephone. The MK-II board converts Bell System's DTMF into binary, and binary into DTMF, thereby making a fully operational Touch-Tone transceiver. On incoming calls, vectored interrupts allow for ring detection as well as detecting the presence of DTMF signaling. This capability permits one to execute programs by calling up his computer and punching buttons on his Touch-Tone telephone. A 4-bit input port allows additional data to be transferred coincident with decoded DTMF. On outgoing calls, digits dialed are loaded into a FIFO buffer at processor speed and unloaded into a DTMF generator at a rate compatible with Bell System's C.O. equipment. A 4-bit output port makes possible the supervision of trunk interface equipment (DAA devices). Single tones may be generated instead of dual tones under software control. Applications of the MK-II include monitoring and tabulating of outgoing phone calls, home security "dialers," and PABX systems. Remote operation of A.C. appliances is also possible by 60-cycle modulation with DTMF signaling. $425.

MK Enterprises, 8911 Norwick Rd., Richmond, VA 23229. (804) 285-2292.

CIRCLE 200 ON READER SERVICE CARD



BUS TERMINATOR

The Exterminator (VTE 100) is a S-100 compatible card that terminates the entire S-100 bus using industrial bridge terminating networks to eliminate crosstalk between busses, overshoots, ringing, and scrambling of data due to interference caused by extraneous noise. As the frequency of any system is increased (2-4 Mhz), "these bus problems become more serious and thus, without termination, the overall system performance can actually decrease." Secondly, the board serves as a card extender for any of your memory and/or I/O cards which may require analysis or maintenance. The Exterminator fuses all extended power busses to protect both the extended card and the power supply from any accidental damage. $51.95.

VAMP Inc., P.O. Box 29315, Los Angeles, CA 90029.

CIRCLE 201 ON READER SERVICE CARD

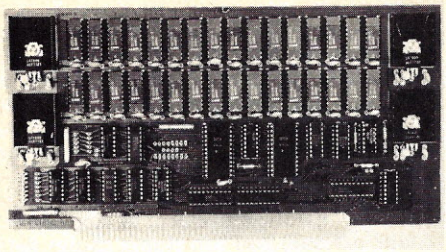
DATA CASSETTE FOR HOME COMPUTING

PerCom Data Company has introduced what is said to be the first data cassette developed exclusively for the home computing user. The PerCom data cassette, designated the Pilon-30, incorporates features "normally found only in high-priced digital cassettes." Principal among the design features of the Pilon-30 is an extra large pilon-coated pressure pad that replaces the fiber pad of ordinary audio cassettes. The Pilon-30 pad provides more uniform tape-to-head contact, eliminates the lint-producing fiber pad that is a source of "drop-out" data error in ordinary cassettes, and assures smooth movement of the tape because of the low-friction pilon coating. Erratic movement of tape across a tape pressure pad sometimes causes data-garbling high-frequency flutter. Another cause of flutter has been circumvented by replacing the pad leaf spring used in ordinary audio cassettes with an energy-absorbing foam spring. The 150 feet of tape of the Pilon-30 cassette was chosen as having an optimum capacity for home computing applications. Data storage is 50,000 bytes of 30-byte-per-second data or 200,000 bytes of 120-byte-per-second data. PerCom Pilon-30 cassettes may be purchased from local home computer dealers or ordered directly from the factory. The minimum direct mail order is $12.45 (five units) plus shipping.

(PerCom Data Company, Inc., 318 Barnes, Garland, TX 75042. (214) 276-1968.

CIRCLE 202 ON READER SERVICE CARD

Soon to be a major motion picture

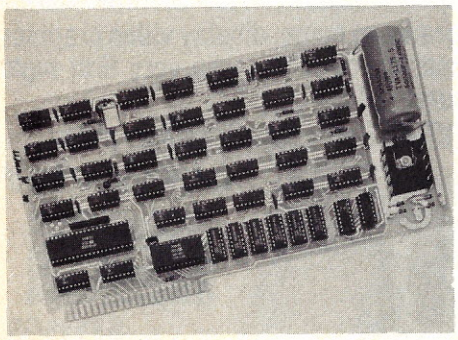


16K STATIC RAM WITH PAGING

Digital Micro Systems has a 16K static RAM for the S-100 bus that uses the industry standard 2114 memory chip and has many extras. The board is completely static, "having none of the timing incompatibility problems associated with dynamic or clocked chip select 'static' RAMs." This means that the DMS board will run with any S-100 system including DMA systems like the 16-bit Alpha Microsystems AM-100. It also runs on Z-80 systems at the full 4-Mhz clock rate. The board features individually addressable 4K blocks, software write protection in 4K blocks, and a paging or block select feature. This allows memory expansion beyond 64K and permits the implementation of low software overhead timesharing systems. $525 for the 16K kit, $295 for the 8K kit, $595 assembled. The board is also available with all but the 2114s for $85 or as a blank board with the manual for $35.

Digital Micro Systems, Box 1212, Orem, UT 84057. (801) 224-2102.

CIRCLE 203 ON READER SERVICE CARD

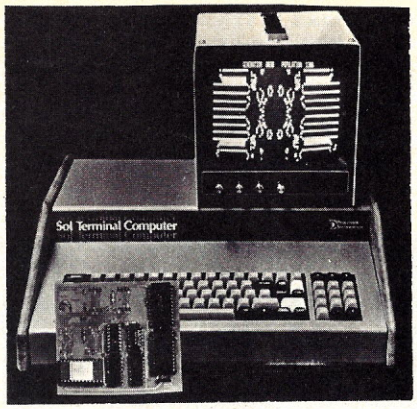


TERMINAL BOARD

RHS Marketing has an Economical Stand-Alone Terminal Board. The ESAT-100 comes either as a kit or assembled and tested. Both versions include on-board regulated power supplies. All that is needed to make it operational are a a 5-V keyboard and a video monitor. Data I/O is serial asynchronous, 11-unit code, TTL compatible. Baud rates are adjustable 300, 600, 1200, 2400, 4800, 9600. Display is 32 characters per line, 16 lines, 2 pages. Also available is an optional piggy back conversion board that will expand the ESAT-100 to 64 characters per line by 16 lines. Features of the ESAT100 include full cursor control, functions of backspace, forward space, line feed, reverse line feed, home, return to end of line. $185 kit, $239 built and tested.

RHS Marketing, 2233 El Camino Real, Palo Alto, CA 94306. (415) 321-6639.

CIRCLE 204 ON READER SERVICE CARD

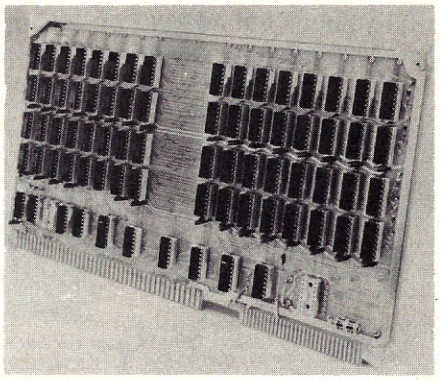


GRAPHICS BOARD FOR SOL

KEA Micro Design of Toronto has announced the GraphicAdd, a piggyback board kit for use in SOL computers and VDM-1 Video Display Modules. GraphicAdd gives graphics capability to these units (128 H by 48 V) and includes a SOL ready-to-load software package. The GraphicAdd board mounts directly on the VDM and SOL main boards. It works by replacing half of the inverse video character set by bit-mapped graphic cells. Thus alphanumerics and graphics can be mixed on the same line. Mode-control options include fixed graphics, switch-selectable graphics, or programmable graphics mode. The software package contains a graphics driver, BASIC Links, LIFE, and demonstration programs $50.

Available from SOL dealers or from Micro-Ware Limited, 27 Firstbrooke Rd., Toronto, Ont., Canada, M4E2L2.

CIRCLE 205 ON READER SERVICE CARD

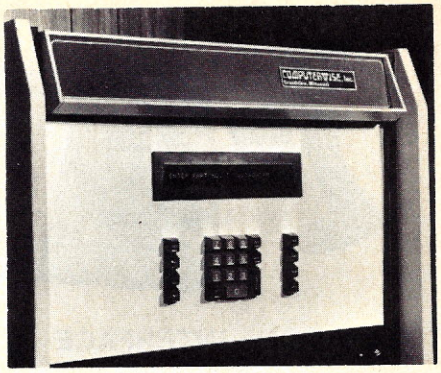


STATIC RAM WITH BATTERY BACKUP

Two static RAM boards capable of battery backup are compatible with Intel's SBC 80/05, SBC 80/10 and SBC 80/20. The RAM-4L contains 4K bytes of RAM, the RAM-8L contains 8K bytes. The RAM-8L uses a single 5V power supply and draws 1.2 amp typical, 1.7 max. under operation. During battery backup at 1.7V, the battery current is .5 amp typical, .8 max. That means that three D-cell alkaline flashlight batteries could back up 8K bytes of RAM for 11 hours. RAM-4L, $312; RAM-8L, $428.

Richard E. Van Antwerp, Electronic Solutions, Inc., 7969 Engineer Rd., San Diego, CA 92111. (714) 292-0242.

CIRCLE 207 ON READER SERVICE CARD



RACK-MOUNTABLE KEYBOARD/DISPLAY

Computerwise offers a rack-mountable keyboard/display unit for use in computer controlled machines, automatic testers and similar applications. The unit can be attached to any computer or microprocessor using an asynchronous RS-232 or 20-mA current loop I/O port. Switches allow the user to select the operating mode including; 110-9600 baud rate, full or half duplex, even/odd/no parity, 5 to 8 data bits and one or two stop bits. The unit mounts in a standard 19-inch-wide equipment rack and requires 10½ inches of panel height. The display provides a single line of up to 32 alphanumeric characters. $750.

Computerwise, Inc., 4006 East 137th Terrace, Grandview, MO 64030. (816) 765-3330.

CIRCLE 208 ON READER SERVICE CARD

TERMINALS

VIDEO TERMINAL KIT

CYBERNEX LIMITED announces the new LTL-1K video terminal kit, featuring a 24-line, 80-character display on a highly legible 9-inch CRT. The LTL-1K controller card includes the power supply components, RS232 interface and keyboard interconnection. The terminal operates at 8 selectable baud rates from 110 to 19,200 baud. Sockets are used for all IC's. Cybernex's fully controllable block see-through cursor features cursor up,

down, forward, backspace, home, home and clear, return to start of line and clear to end of line controls. Kit with keyboard, $695; without keyboard, $525. Assembled and tested form with keyboard, $895.

Cybernex Limited, 3221 Council Ring Road, Mississauga, Ontario, Canada L5L 1L5.

CIRCLE 209 ON READER SERVICE CARD

libraries. The diskette containing the macro assembler (machine code only) is $70 (diskette order must be accompanied by the purchaser's CP/M serial number). The documentation is available separately for $15 (no serial number required), with the option of later diskette purchase at $60.

Digital Research, Box 578, Pacific Grove, CA 93950. (408) 373-3403.

CIRCLE 210 ON READER SERVICE CARD

MITS BASIC (3.1). Two versions of the program on one 60-minute cassette play a challenging game that conforms to International Rules. The first version requires a total of 16K of memory, inclusive of 8K BASIC. The second version is more graphic and requires an additional 4K. The checkerboard is pictorially displayed on the CRT. As the player and computer each take turns, the checkers blink and move to indicate their passage. Kinged pieces are identified on the display and messages appear at the right of the board relating to each move. In accordance with International Rules of the game, the program will not accept illegal moves and warns of their entry. $10.

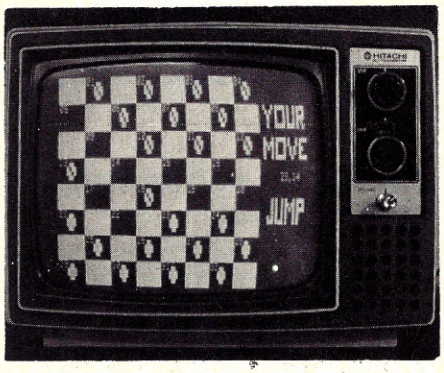
Compu-Quote, 6914 Berquist Ave., Canoga Park, CA 91307. (213) 348-3662.

CIRCLE 211 ON READER SERVICE CARD

SOFTWARE

CP/M MACRO ASSEMBLER

A new macro assembler called MAC is offered by Digital Research. MAC operates with the Digital Research standard CP/M Diskette Operating System and implements the recently redefined Intel standard macro facility, while retaining upward compatibility from previous standard assemblers. Specific features of the new macroprocessor include conditional assembly (IF, ELSE, ENDIF) with assembly-time expressions (+, -, *, /, MOD, SHL, SHR, AND, OR, XOR, HIGH, LOW, LT, LE, EQ, NE, GE, GT, and NUL). Repetition of source statements is provided with indefinite repeat on character (IRPC), indefinite repeat on text (IRP), and numeric repeats (REPT). Parameterized macros are stored using the MACRO statement which can appear in the mainline source program or be called out from previously defined macro



VIDEO CHECKERS

Compu-Quote has developed several games on cassettes, recorded in the Tarbell format and programmed in MITS BASIC. Contained on one cassette is VIDEO CHECKERS, which produces checkerboard graphics on the CRT when used with the Polymorphic Video Interface and 64-character option. The game plays under

SOFTWARE LIBRARY

SOFTWARE Ltd. announces the availability of a BASIC language library of programs ready to run on North Star disc media. These programs are "bug-free" and ready to run. Over 45 programs, including business, finance, family budget and games are immediately available. Most programs are priced from $2 to $5 each. The library includes; STARTREK, STARLANES, CHECKBOOK, FAMILY FINANCE Etc.

SOFTWARE Ltd., Box AF, Woodbridge, CT 06525

CIRCLE 212 ON READER SERVICE CARD

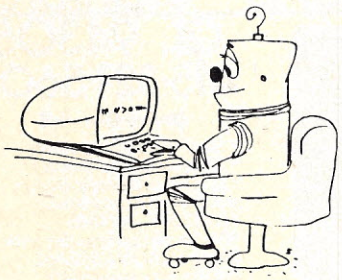
CIRCLE 123 ON READER SERVICE CARD

MACRO-ASSEMBLER

CHROMOD Associates has developed SMAL/80, a compiled, structured, macro-assembly language for 8080 and 8085 microprocessors that requires only 7K of memory. SMAL/80 statements are written in a symbolic notation resembling PASCAL and PL/M that simplifies considerably the writing of assembly language programs. SMAL/80 also incorporates the basic structured-programming constructs, the DO-END, IF-THEN-ELSE, and LOOP-REPEAT, which may be combined with and/or nested within each other without limit to form highly complex statements. The code produced by the compiler is as efficient as that written in a traditional assembly language by a skilled programmer. The SMAL/80 package includes a 2K macro preprocessor written in SMAL/80 that greatly extends the usefulness of the language. The macro preprocessor permits conditional expansion of statements, unlimited nesting of macros, and has a natural notation that is conducive to efficient, error-free programming. This extensible microprocessor language "combines the operating speed and efficiency of a traditional assembly language with the convenience, logical power, and versatility of a high-level language." Programmers can now write complex 8080 and 8085 programs "with the same ease and assurance with which they now write high-level programs, and without having to pay any penalty in superfluous code or reduced speed of execution." $75.

Chromod Associates, P.O. Box 3169, Grand Central Station, New York, NY 10017.

CIRCLE 213 ON READER SERVICE CARD

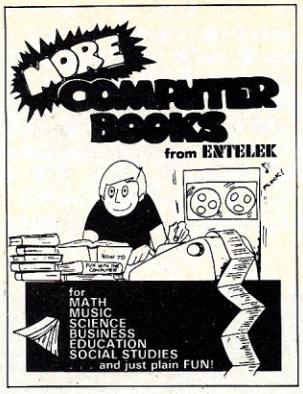


MAILING-LIST PROGRAM

From Williams Radio and TV, the ML-INS program package for the North Star Disk System (SWTPC disk version available soon) is a comprehensive mailing list program package, a modular program set which enables the user to start and effectively maintain one or more mailing lists. Operations include: Add, Delete, Search, Sort, Auto-Sort, and Sequential Printout. Features include: user-selectable defaults for ease of entry, user-selectable number of labels across page for different printers and label sheets, and user-selectable 3 or 4-line address for each independent entry. With complete documentation and North Star diskette, $25. Documentation package only, $4.50.

Williams Radio and TV, Inc., Computer Division, 2062 Liberty Street, P.O. Box 3314, Jacksonville, FL 32206.

CIRCLE 214 ON READER SERVICE CARD



COMPUTER PROGRAMS IN BOOK FORM

A new catalog from ENTELEK lists over 40 computer books for math, music, science, business, education, social studies ... and just plain fun. Many of the books listed contain computer programs in BASIC and APL. There is, for example, a series of 52 programs in math, a series of 14 in physics, a series of 8 in genetics, and so on. One book contains an APL program to generate math tests. The skills tested include addition, subtraction, multiplication and division of whole numbers, decimals, fractions and mixed numbers, and per cent calculations. The math test generator, used at Illinois State University, saves the teacher many hours.

ENTELEK, P.O. Box 1303, Portsmouth, NH 03801.

CIRCLE 215 ON READER SERVICE CARD

DISASSEMBLER FOR NORTH STAR USERS

DIS resides at 2A00H just as North Star BASIC does, and is completely integrated with the DOS from North Star for all I/O. Commands include both ASCII and hexadecimal dumps of memory, cross-reference symbol table, listing in format and free-format forms, and a "quick-look" variation for rapid decision making disassembly. Unique to this disassembler is the ability to leave a file in RAM for the assembler portion of XEK to re-assemble. This feature is invaluable when relocating a program to a different area of memory. When disassembling a program, DIS builds a symbol table and then cross-references it. $48.

Byte Shop of Westminster, 14300 Beach Boulevard, Westminster, CA 92683. (714) 894-9131.

CIRCLE 216 ON READER SERVICE CARD

SUPER BASIC

Technical Design Labs is introducing Version 3.0 of Super BASIC, a 12K BASIC interpreter, which up-grades and supersedes Version 2.1 with numerous speed and error-handling features. Version 3.0 provides programmable error-handling that allows the user to specify special error-handling routines processing any error occurring in the basic program without aborting the program. Version 3.0 allows for the serial input and output of data from the Zapple Monitor defined reader and punch devices. This product has a dynamic load/save specification; Version 3.0 allows the program identifier in a "LOAD", "LOADGO", and "SAVE" command to be an arbitrary string expression. Version 3.0 of Super BASIC is being released under CP/M Version first and later as a serial paper tape version. It occupies 12K of core. Although primarily designed to run on TDL's Z80 microcomputer system, it is adaptable to other Z80 based systems. Super Basic Version 3.0 is on a diskette and is a part of TDL's Software Package A which consists of Version 3.0, the Macro Assembler 2.2, Z-TEL Text Editing Language and the Text Output Processor. $249.

Contact Barbara Greenbaun, Director of Public Relations, Technical Design Labs, Research Park, Building H, 1101 State Road, Princeton, NJ 08540. (609) 921-0321.

CIRCLE 217 ON READER SERVICE CARD

MISCELLANEOUS

HOLE PUNCH

This precision steel punch "exactly duplicates the punched holes on standard computer cards. Create your own, or correct mistakes." Instructions included. $5.50.

Punch, Box 727, Stratford, CT 06497.

TRS-80 MAGAZINE ON CASSETTE

CLOAD Magazine is something new: it is written especially for Radio Shack's TRS-80 computer; directly for the computer, that is. It is "printed" on a standard audio cassette and will load directly into the TRS-80 computer. Its "articles" are really programs ranging from short games to involved programs of a practical nature. Emphasis will be on education, both "tutor" style and through games. People who have programs can submit them for publishing much as the general author submits articles to a regular magazine. There will be 12 issues a year, each issue consisting of an audio cassette with six to ten programs (more, if possible), an index, and an instruction sheet. Charter subscription rates are $24 for one year (12 issues).

CLOAD Magazine, Box 1267, Goleta, CA 93017.

CIRCLE 218 ON READER SERVICE CARD

Radio Shack's personal computer system?
This ad just might make you a believer.

You can't beat the 4K system at $599

TRS-80 "Breakthru"
- TRS-80 microcomputer
- 12" video display
- Professional keyboard
- Power supply
- Cassette tape recorder
- 4K RAM, Level-I BASIC
- 232-page manual
- 2 game cassettes

...or the step-up 16K system at $899

TRS-80 "Sweet 16"
- Above, except includes 16K RAM

...or the fast 4K/printer system at $1198

TRS-80 "Educator"
- Above, except includes 4K RAM and screen printer

...or the Level-II 16K/printer/disk system at $2385

TRS-80 "Professional"
- Above, except includes 16K RAM, disk drive, expansion interface, and Level-II BASIC

So how are you gonna beat the system that does this much for this little? No way!

...The amazing new 32K/Level-II/2-disk/ line printer system at $3874

TRS-80 "Business"
- Above, except includes 32K RAM, line printer, and two disk drives

Get details and order now at Radio Shack stores and dealers in the USA, Canada, UK, Australia, Belgium, Holland, France, Japan. Write Radio Shack, Division of Tandy Corporation, Dept. C-005, 1400 One Tandy Center, Fort Worth, Texas 76102. Ask for Catalog TRS-80.

Radio Shack®
The biggest name in little computers

Radio Shack TRS-80

User's Manual, Applications Programs

Stephen B. Gray

User's Manual

The "User's Manual for Level I" wasn't quite ready at the same time Radio Shack's TRS-80 microcomputer system was, so the first couple of thousand purchasers got a 30-page "Preliminary User's Manual." By now, all 4,000 or so should have received the regular User's Manual.

The preliminary manual, although very well written, didn't have space to go into detail, and could only teach the basics of BASIC. The User's Manual goes into great detail in its 233 pages, and does such a fine job of teaching BASIC that Radio Shack has decided to sell the manual separately, at $9.95. The only non-"standard" features of Level I BASIC are the graphics statements and the "Shorthand Dialect" that permits using P. instead of PRINT, D. for DATA, N. for NEXT, etc., which allows packing more than the usual number of statements on a CRT line. Apparently no other BASIC uses such abbreviations, other than Palo Alto Tiny BASIC, around which TRS-80 BASIC was designed.

The User's Manual consists of several introductory pages on setting up the system, 26 chapters on BASIC that range from using PRINT to debugging, 24 pages of answers to the programming exercises given in the chapters, 16 prepared user's programs, and appendixes that provide user subroutines for trig and other functions not provided in the LEVEL I ROM, information on using cassette data files, and a program the user can enter on his TRS-80 to check out the functions, the RAM memory, and the display (with a test pattern).

One of the first things noticed on opening the User's Manual is the four-inch righthand column that has a few illustrations and explanatory notes in it, but which is rather sparse. The manual, when opened, is over 22 inches wide; without that column of notes, the manual could be only about 14 inches wide when opened, and the notes, when incorporated into the text, would make it only about 16 pages longer.

However, there's a good reason for the notes column, which was written as footnotes to the text by Radio Shack's chief technical writer. As the manual goes into successive editions, the notes column will contain more and more information, as TRS-80 owners write in to ask for clarification of certain points, for instance.

Page 1 consists of a prefatory "Personal Note from the Author," Dr. David A. Lien, who writes that "This book is written specifically for people who don't know anything about computers, and who don't want to be dazzled by fancy footwork from someone who does. It is written to teach you how to use your Radio Shack TRS-80 computer and start you on a fast track to becoming a competent programmer. To that end, every fair and unfair, conventional and unconventional, flamboyant and ridiculous technique I could think of was used. I want you to have fun with your computer! I don't want you to be afraid of it, because there is nothing to fear." He then goes on to say, "The only restraints put on this book were good taste and a genuine attempt not to insult your intelligence. Beyond that, it contains no 'snow jobs,' no efforts to impress or intimidate you, and no attempt to sell you anything except the idea that computers are just not all that hard to use." Three more sentences are worth quoting, "The real enjoyment begins when your imagination starts the creative juices flowing and the computer becomes a tool in your own hands. You become its master — not the other way around... Enjoy your new computer!"

One of the first things I realized, after reading a chapter or two, is that the author must have taught BASIC, because there is all sorts of information in this book that an author learns only when interacting with a class. Having read over 50 books on BASIC (and reviewed 34 of them in a six-part series in *Creative Computing*), I knew this was no sterile text written by someone who assumed his readers were as clever as himself. Just about all the bases are covered here, and with the added notes in the righthand column, very little gets left out. (Dr. Lien, incidentally, is dean of Grossmont College in San Diego, California, and *has* taught BASIC, for some years.)

There are some peculiarities, however, The closest thing to an index is the table of contents on page 3. To be sure, the "Summary of LEVEL I BASIC" on the last two pages does give the chapter in which each command, statement, operator or function is described, but only by chapter number, not by page. So you still have to refer to page 3. And the chapter headings appear only on page 3; not a single one is repeated at the beginning of the chapter itself, so if you look at the first page of any chapter, there's no indication, as there should be, of what the chapter contains. This is a small point, however, when considering that the average user of a TRS-80, once he's gone through the manual, may seldom refer to it again except to the summary, to see how to set up a seldom-used statement or function.

The language of the manual is conversational, comfortable, and often colloquial. To teach a point about the BASIC interpreter, the author deliberately leads the reader into making an error, and then writes, "Oh — sorry about that! It 'bombed,' didn't it?"

The pace is slow and easy; the first program is a one-line 10 PRINT "HELLO THERE, I AM YOUR NEW TRS-80 MICROCOMPUTER!" followed by two pages of explanatory text about PRINT, NEW, RUN, correcting errors, and the ENTER and BREAK keys.

The last note in this chapter, in the righthand column, is a "Special message for people who can't resist the urge to play around with the computer and skip around in this book. *(There always are a few!)"* and goes on to tell how to regain control of the computer if lost, by using the BREAK key, or the Reset button inside the left rear corner.

As an indication of the book's thoroughness, and the author's competence, page 13 shows how, after using 99 END, you can erase this line by typing just 99. Then you're told how END can be put on the highest possible line number, 32767, and the program runs exactly as before. Then you're asked to change the END line number to 50000, which will result in the error

message "HOW?", and which is then explained in detail.

The notes column contains dozens of anthropomorphic computer cartoons; drawings of a computer "person," with video monitor as head and keyboard as torso. A nice touch.

By page 21, the reader is asked to write programs, four of them: simple five-line programs.

The section for FOR-NEXR provides an excellent grounding in its use, although not much space is devoted to explaining print zones, the use of semicolons, order of operations and the use of parentheses. Perhaps later editions will, if enough readers ask for more information.

The author recommends indenting FOR-NEXT loops to make "reading and troubleshooting easier." He also uses the phrase "fall through," which very, very few books on BASIC do, although the phrase is common enough in the field.

As a further example of the manual's informal style, the explanation of multiple-statement lines says, after the first example, "Egad, Igor — we've created a monster! *Will it work?* RUN and find out."

The chapter on the INTEGER function notes that a rounding error occurs when separating the integer and decimal portions of 3.14159, and then puts in a commercial for the LEVEL II BASIC by saying that users who have it will not notice this routine rounding error. Then it adds, "If we solved **all** the world's problems with the bottom-of-the-line machine you might not want to upgrade to the higher power model, and one doesn't stay in business long that way, does one?"

Page 74 gives some fine troubleshooting advice, such as inserting "temporary PRINT lines anywhere in any program so we follow every step in its execution" to "observe the inner details of the calculations." And then, "It is most helpful of all when inserted in FOR-NEXT loops."

Page 79 offers a "trade secret" I don't remember seeing in any other book on BASIC: adding .2 to N, making ON N+.2 a way of taking care of rounding errors, since the ON-GOTO statement contains its own INT function.

Page 99 notes that RND(O) gives random values between 0 and 1. (In my introductory article in the Jan-Feb issue, I forgot to include this, and said only that RND(10) provides random integers between 1 and 10, inclusive.)

The text about the graphics "light" says it is 2 dots wide by 8 dots high. Actually, there aren't really any "dots," just horizontal raster lines, and the ratio is much closer to 3:7.

The Video Display Worksheet, on page 106, is for laying out both graphics and interspersed alphanumerics, although very little information is given on the use of this worksheet. The proportions of the worksheet don't correspond to the video-monitor display; on the worksheet, the proportions of the "light" are in the ratio 1:2, which results in the worksheet being 8-5/16 inches wide, while the display (on my TRS-80, at least) being only 7-5/8 inches wide. Because the vertical measurements are almost the same on the worksheet and on the screen, the resulting distortion means that whatever you design on the worksheet won't look quite that way on the screen...

Several of the graphics programs use a RESET line before a SET line, without any explanation of why you're trying to reset a light "block" that just doesn't seem to be there yet.

What may be the only major error in the book is on page 118. The program won't run at all unless the minus sign in line 110 is changed to an equal sign, making it 110 IF Y=48 THEN 130.

Cover of the Radio Shack TRS-80 User's Manual for Level I BASIC. The holes on the left margin are for the plastic binding that allows the open book to lie flat.

Page 121 contains the only sentences I would argue with: "Because the ideas come so fast in the area of graphics, we have deliberately chosen to show you a lot of examples without getting bogged down in detailed explanations of how each one works. There is no substitute for lots of experimenting with graphics, and you know the basics. Put in your time, study the examples, and soon you can apply for membership in the artists' "guild." I'd still like to know about RESET before SET.

The chapter on flowcharting is not very good; it should have taken some programs previously demonstrated and flowcharted them, which would be much more meaningful than to flowchart a new program.

Chapter 25 on Advanced Subroutines demonstrates the use of the trig and exponential subroutines given in Appendix A because there was no room in the ROM for them. The formulas used in the subroutines were modified to fit the Radio Shack TRS-80, according to the Radio Shack technical writers, and can be found in any mathematics manual, such as the McGraw-Hill *Mathematics Manual* by Frederick Merritt. The trig and inverse trig functions are derived from Taylor series, the SQR from the Newtonian approximation, and the exponentiation formulas were found in *Interface Age* (Feb. 1977, p. 103). Some "interesting shortcuts" were found in *Scientific Analysis on the Pocket Calculator,* by John Smith (Wiley, 1975).

The last chapter, on debugging, is excellent, one of the best in the book, and offers some very good advice, including the unexpected, "By the way... a one-semester course in beginning typing can do wonders for your programming speed and typing accuracy."

At the end of the text is a fine sendoff, "Beware of Creeping Elegance," meaning that "it's easy to lose sight of the purpose of the program."

The User's Programs include some interesting ones, such as a 12-hour clock and the Parker Brothers game of Sorry, some useful ones such as loan amortization and speed reading, and a couple of curiosities such as the long "Design Program for Cubical Quad Antenna."

All in all, except for not enough information in a couple of areas such as graphics, this is a manual with much valuable advice that would apply equally to commercial uses of BASIC as well as to hobby uses. Here is a firm grounding in the highways and byways of BASIC, by an excellent teacher.

Educational Systems: Math I

The first of the Radio Shack applications programs reviewed is Math I, consisting of a three-ring binder with 28 pages of Teacher's Guide, and three cassettes mounted inside the front cover in a simple plastic holder. This "3-cassette portfolio" is $19.95.

Math I teaches the four basic math operations, with emphasis on *repetition and review.*

The first 3½ pages explain the system, which presents the material to the learner in several variations:

3	7	4	7
+4	-4	+3	-3
7	3	7	4

This is called a *fact set,* which is "formulated this way to show relationships of math functions. The learner associates and memorizes easier."

Half a page explains, in 10 steps, how to load the cassette program into the TRS-80 computer and run it. The first thing on the screen is:

MATH I — ADDITION AND SUBTRACTION
WHAT WAS THE LAST GROUP COMPLETED?

This may throw the beginner, until he realizes that if he hasn't completed any

group at all, and the computer insists on a response, then the only possible response is zero. It works, and YOU WILL NOW BEGIN GROUP 1 comes up on the screen, soon to be replaced by the first fact set.

The first fact set is already worked out, with answers, and when the learner has memorized the relationships, he presses ENTER, which repeats the fact set but without the answers. He has only to copy the answers from the fact set above; if he makes a mistake, the computer urges him to TRY AGAIN. When he succeeds, VERY GOOD appears on the screen at top right, and at bottom right appears a rocket, which slowly lifts off, moves skyward, and disappears at the top of the screen.

The same four-item fact set is presented again, in a slightly different order: the second and fourth items are switched. Four correct responses again causes a lift-off.

This fact set is repeated, after the already-completed set is deleted from the top of the screen. Now the learner is on his own, with nothing to copy from, and if he's learned the lesson, a third rocket goes up.

The computer moves on to a second fact set and repeats the same sequence as with the first set, complete with three rockets, Then both sets are presented together, eight items; first a set of eight with correct answers, with a set below to be filled in. If done correctly, the result is:

YOU HAVE NOW COMPLETED GROUP 1
VERY WELL DONE!!!!

IF YOU WANT TO STOP, TYPE 11
IF YOU WANT TO GO TO THE NEXT GROUP, TYPE 99?

And so on for 23 groups, by the end of which the learner has worked his way through all the add and subtract relationships of the numbers 1 through 10, plus a few more. All 23 groups are shown in the manual, covering 4 pages.

If a learner goes through all 23 groups, he sends up a rocket six times in each group. Each rocket takes 11 seconds to go up, for a total of 1518 seconds, or 25.3 minutes spent watching rockets go up. The minimum time to go through all 23 groups is about 40 minutes, so a really fast learner, making no mistakes, would spend over 60 percent of his time in rocket-watching. But few learners could go that fast, so an average learner may spend only 10 or 20 percent of his time watching those 138 rockets loft skyward, which isn't much time if he's really into rockets...

Multiplication and division, on the second cassette, is about the same, with the same six rockets per group. The same program is repeated on the latter part of the tape, as it is on the first cassette. What's on the other side of the tape? The same thing: you get four recordings on each cassette, of the same thing.

All 23 multiplication and division groups are given in the manual, on 8 pages, followed by 2 pages that present a detailed "Preparation Before Beginning Math I Program," repeating much of the previous text.

After each three groups on either tape, the learner has the option of stopping, going on to the next group, or taking an evaluation test, from the third tape:

YOU ARE NOW READY FOR EVALUATION TEST NUMBER 1
TAKE THE TEST
AND SEE HOW YOU ARE DOING

With this short tape, the learner enters a 1 to be tested on addition and subtraction, and a 2 for testing on multiplication and division. Either number then gives a choice of 8 tests, each containing 16 groups. Your score is calculated; if you get one wrong, YOUR SCORE IS 93.75%. The evaluation tests give an OK for each correct answer, right away, to let you know as soon as possible if you got it right.

If the score is less than 85 percent, meaning if more than two answers are wrong, the learner is urged to GO BACK AND REVIEW THE LAST GROUP.

The last half-page of the manual is a *Summation,* with, among others, these sentences: "It is our belief ... that Math I as taught on the TRS-80 Computer, achieves many skills common to math instruction in the 'old fashioned' manner that is basic instruction as well as incorporation of many innovative ideas used in the 'new math'." These educational programs are, to my very subjective thinking, the best of the four applications packages obtainable at this writing.

Kitchen

Three of the first four applications packages are in binders; "Kitchen" comes in a hang-on-a-pegboard package, with one cassette and a manual that provides a couple of paragraphs of information about the two programs on the tape, *Recipe Conversion* and *Message Center.*

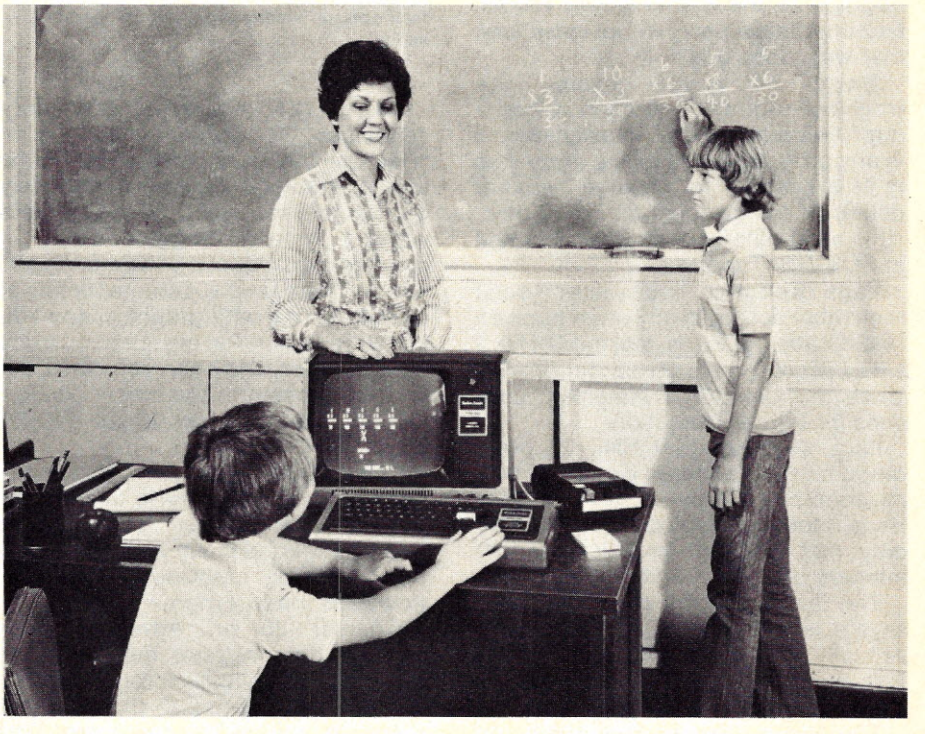
For the first, the manual says "This program converts a recipe for a given number of servings to a recipe for any other number of servings." Loading takes a minute and a half. The program repeats the sentence from the manual, "This program converts..." and then asks you to "Type in the no. of servings in the recipe and the no. of servings desired." After you type in, for example,

4,9

the screen comes up with instructions on what to do next, and when you understand them and press ENTER, the screen changes to show, at the top:

NEW RECIPE (9 SERVINGS)

and at the bottom of the screen you're asked to provide, in sequence, the amount of ingredient, its measure, and its name. As soon as the name is provided, and ENTER pressed, the



Students learning how to multiply through using a Math I program cassette in the Radio Shack TRS-80 microcomputer system.

computer makes a fast calculation and puts, for example

4½ CUPS SUGAR

up under NEW RECIPE. And so on for 13 ingredients. The manual says that when all the ingredients are entered, type 99, and the computer is supposed to "display the complete new recipe." Actually, it's already done this, and typing 99 will ask if you have any more recipes to enter.

If your recipe has 12 or less ingredients, you're OK. But if it has 13 or more, you've got a problem, because if you type in the 13th, it appears briefly but then disappears. The following two also disappear. Although the 16th ingredient does appear on the screen, it wipes out the first one. The answer is to either write down the first 12 ingredients or measure them out in the kitchen, and enter a 99 and then a 1 so as to be allowed to enter another 12 ingredients. If you type 99 and then a 2, you get

END OF SESSION

When listed out, the program turns out to be, as expected, quite simple. If the amount in the new recipe is an integer, the program simply prints that integer. If otherwise the program converts the fractional portion of the new amount, rounds it off, and converts it to eighths.

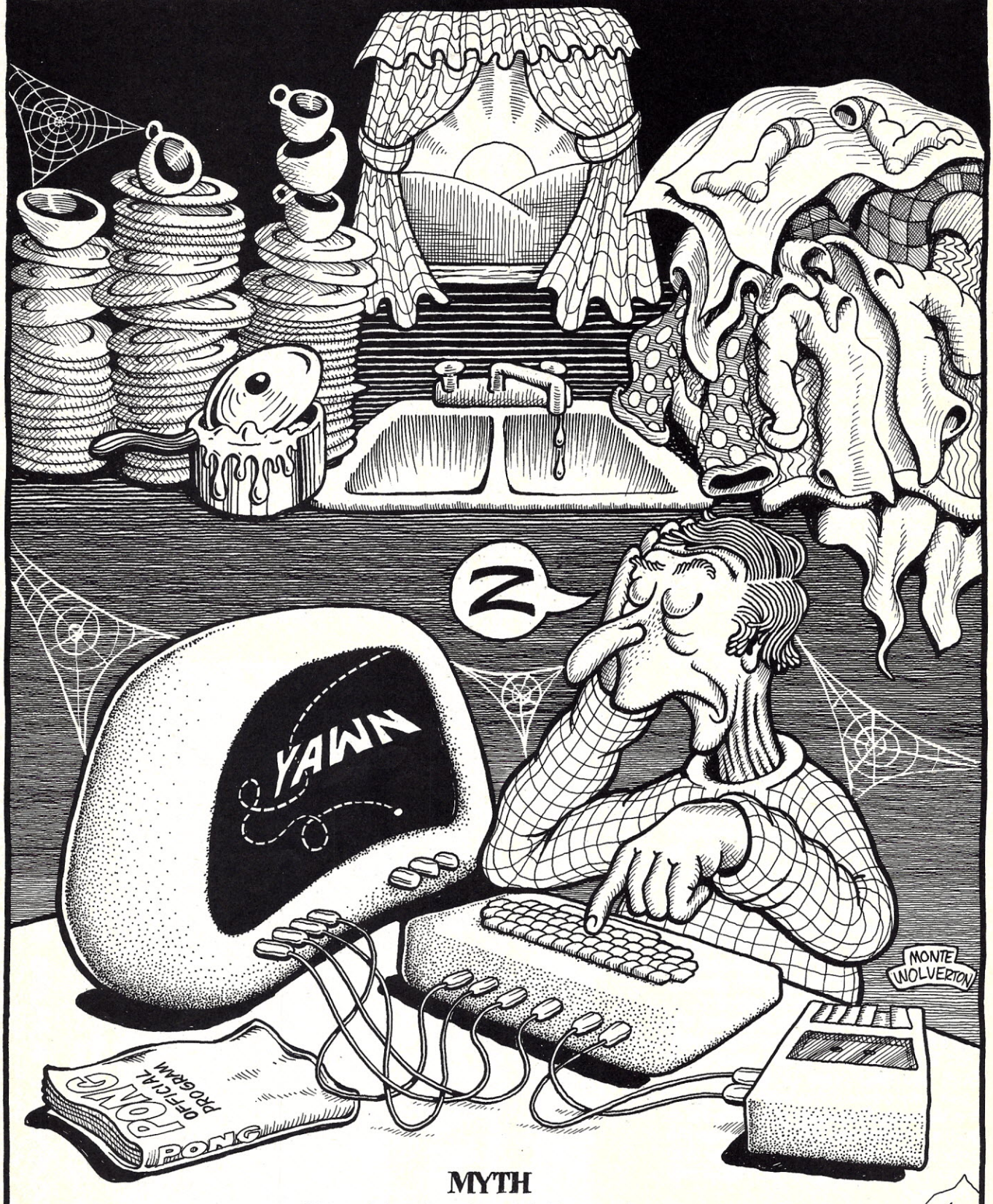
Most of this conversion could be easily (and more cheaply) done on a calculator. All you need is a conversion table for converting fractions to decimals, and use the new-recipe ratio as a constant. However, this computer program does convert to integers plus eighths, which would take more than just a little extra time with a calculator.

On the flip side of the tape is the *Message Center* program, which when entered into the TRS-80 (in 40 seconds) with CLOAD, advises you to type in your message, and at the end of each line to press ENTER. "After you have typed and entered the last line, press Shift/Q." When you press Shift/Q, all it does is to wipe out the messages. So all this does is provide you with an electronic blackboard.

What neither the program nor the manual tells you is that you don't press first Shift and then Q, but hold down Shift and then press Q. Another surprise may come if the user tries to list the Message Center program. He can't, as it's in machine language, since the only way to get such long strings is to write the program in assembly language.

This cassette, at $4.95, doesn't seem destined to be a best-seller at Radio Shack. However, if enough people are interested in converting a recipe for 4 into a recipe for 9, I could be very wrong. After all, it will even convert to a recipe for 9.6, if you want to cook for 9 adults and one small child. ■

COMPUTER MYTHS EXPLAINED



MYTH

IF HOME COMPUTERS **REALLY** CAN'T DO **PRACTICAL** THINGS, LIKE WASHING THE DISHES, VACUUMING THE CARPETS, AND DOING THE LAUNDRY, THEN THEY ARE A POOR INVESTMENT FOR THE AVERAGE PERSON, EXCEPT FOR MAYBE AN OCCASIONAL GAME OF PONG, WHICH WILL QUICKLY BECOME A BORE.

COMPUTER STORE STATEMENT

YOU OWE US PROFUSE AMOUNTS OF MONEY! PAY UP! NOW! OR ELSE! Thank You

Cut out this ad and mail it to us with your resume.

Maybe your job isn't getting duller.

Maybe you're getting sharper.

This is one place your job won't dull out on you. As soon as you're too sharp for it, we have another place for you.

We've grown in ten years to $250 million plus in sales. Data General is the computer company that changed the expression on the face of the industry, with systems like NOVA and ECLIPSE. We're now up to 44,500 in installations worldwide.

And we want sharp people. Tell us all about yourself. Check position which interests you most.

Systems Analysts

You will participate in the development of on-line manufacturing systems, therefore at least three years of directly related experience is required, together with the ability to prepare system design and detail program specification. A working knowledge of Cobol and exposure to assembler is also needed. Knowledge of communications systems and data base management is a plus.

Diagnostic Programmers

You will design, code and debug assembler language programs for fault isolation, to the chip level, in digital systems. You will also write functional level system exercisers for stand-alone and disc-based real-time operating systems. A technical degree or the equivalent experience is needed, together with a thorough understanding of digital system hardware. Hardware troubleshooting experience and 3 or more years' programming experience, preferably in diagnostics is very desirable, especially with strong assembly language skills.

Mail your resume to Tom Aldrich at Data General Corporation, Route 9, Southboro, MA 01772. Data General is an equal opportunity employer, M/F.

My resume's attached. I want to turn a dull ache into a sharp ambition.

Signature: _____

Data General

Beginner's View of the SAM76 Language

Claude A. Kagan

For a long time, BASIC and assembly language have been the only programming languages available for microcomputers. But already we're seeing the development for microcomputers of more of the traditional computer languages— FORTRAN, COBOL, APL, etc. However, some languages not widely available on any computers before are now being released to the amateur computing community. The following article describes a new language called SAM76. SAM is a "string processing macro language" based on two other macro languages: Strachey's GPM and McIlroy's M6. SAM76 is not a typical language utilizing line after line of code to be executed in sequence, and different types of variables. Rather, SAM76 is designed to handle "texts" that may contain procedures to be executed, or just data to be manipulated. Thus SAM excels at text processing. In fact, subsets of BASIC, ALGOL, and APL have been easily implemented in SAM (though they suffer in executional speed because you have an interpreter interpreted by an interpreter.... On the other hand, SAM's built-in math functions are rather crude: only add, subtract, multiply, and divide in integer mode only. But since numbers are stored just like other texts (a string of digits) it is simple to do interesting things like calculate 100-digit factorials. Of course it is simple to add your own number-processing functions to SAM.

As of this writing, distribution plans for SAM76 are not definite. The language and documentation will be in the public domain, so probably you'll be able to get a copy of SAM76 through local computer stores, or computer-club software libraries. SAM76 is available for 8080 and Z-80 microprocessors (the Z-80 version fits in 8K, and the 8080 version is about 1K larger). For further information on SAM76, and some helpful information on loading and running it, see the January 1978 issue of *Dr. Dobb's Journal.* Also, we'd be interested in readers' reactions to this language. —Steve North

Beginner's - Part I - Operation and Syntax

The SAM76 language deals mostly with the manipulation of text. It is designed for use through a reactive machine such as a personal computer such as a "home reckoner" set.

The language design has the structure to allow interaction of functions resident in the machine with expressions, scripts or procedures written by the user; in this manner the language gives the user an unusual amount of flexibility and freedom for invention and extension.

The syntax consists first of a "warning character" followed by the expression itself then terminated by a different second "syntax marker"; in the following discussion the "warning characters used will be one of the following three: % - percent sign, & - ampersand or ! - exclamation mark; the "syntax marker" will be the / - slant sign for example:

```
%...../ or &...../ or else !...../
```

The foregoing three examples represent respectively the three types of expressions used in the SAM76 language and are known respectively as "active", "neutral" or "protected" expressions; the significance of the three types will be explained later.

The expression itself is made up of arguments which are separated by commas. The first argument designates the action to be taken. If this first argument consists of two or three alphabetic characters, the action to be taken may well be one defined as a function built in to the language or otherwise a language primitive function. Each argument following contains text or data to be dealt with by the action taken within the execution of the expression.

For instance we wish to add two and four; consequently we type everything in the following example up to and including the "=" equal sign which tells the computer to do its thing:

```
{}----------------
{}  %ad,2,4/=6
{}
--------------------
```

The two letter code "ad" signifies the primitive of addition. Upon execution, which was initiated by the equal sign after the slant sign, the value of the second argument or 2 was added to the value of the third or 4. Then the value computed is outputted. The system then returns to a waiting condition known as the idling program which identifies itself by moving the "cursor" or printer to the beginning of the next line.

The idling program is actually the following expression:

```
%os,%is//
```

When starting, the innermost expression is located which contains an "is" primitive; "is" - or "input string" accepts input from the keyboard up to the reception of the current "activator" namely (in our case) the equal sign. The computer replaces the %is/ expression with this typed in text. Now the system goes back one level of nesting to the expression whose command was "os" or "output string"; this expression outputs the contents of the second argument which is now the text accepted from the keyboard, thus repeating what was typed in. For example:

```
{}------------------------
{}  %os,%is//=ABC=ABC
{}
------------------------
```

In actual fact the expression "%os,%is//=" is executed everytime the idling program is loaded; it is not printed out and lives in what is known as the working area of the memory so actually the printed example should be:

```
{}-------------
{}  ABC=ABC
{}
-------------------
```

It is important to be able to store text, script or procedures in memory. To this end the "dt" which is the mnemonic for the "define text" function is used. If we wish to define a text to be named "A" containing the words "AN APPLE" we type:

```
↑↑-----------------------
{}  %dt,A,AN APPLE/=
{}
-----------------------------
```

Now stored in memory is a "text" named "A" containing the words "AN APPLE". To retrieve this information we "fetch" the "text" named "A", and in this processs the second argument of the idling program will contain the words stored and the "os" will output the value returned in the fetching of "A" thusly:

```
↑↑---------------------
{}  %ft,A/=AN APPLE
{}
-----------------------
```

When we defined the text "A" nothing was returned since "dt" does not return any value on execution.

To Continue - "pt" or "partition text" removes one or more characters from a string and in its place sets markers which represent the value of the partition.

```
↑↑---------------
{}  %pt,A,AN/=
{}
----------------------
```

The second argument holds the name of the text to be dealt with; the third argument is the string of characters which if found in the "text" will be removed and replaced by partitions. Now to examine "A":

```
↑↑-----------------------
{}  %ft,A/= APPLE
{}
-------------------------
```

Note that "AN" is missing and nothing shows its presence because the expression that fetched "A" above did not require any partitions to be "plugged" in a manner to be shown later.

We will now define another text to be named "B":

```
↑↑-------------------------------------
{}  %dt,B,THE SHACK ON THE HILL/=
{}
---------------------------------------
```

We partition that text on space:

```
↑↑-----------------
{}  %pt,B, /=
{}
----------------------
```

We fetch "B" and get:

```
↑↑-----------------------------
{}  %ft,B/=THESHACKONTHEHILL
{}
----------------------------------
```

Notice the spaces are omitted.

"fe" or "fetch element" returns the contents of the text designated by the second argument; but on finding a partition it stops output. On the execution of the next "fe" on that text the next element of the text between partitions are returned:

The first:

```
↑↑-------------------
{}  %fe,B/=THE
{}
----------------
```

The second:

```
↑↑---------------------
{}  %fe,B/=SHACK
{}
--------------------
```

The third:

```
↑↑----------------
{}  %fe,B/=ON
{}
----------------------
```

It is very simple to find out where the partitions and the divider happen to be at any time by using the "vt" or "view text" primitive thus:

```
↑↑---------------------------------------------
{}  %vt,B/=THE[1]SHACK[1]ON[1][|]THE[1]HILL
{}
--------------------------------------------------
```

In this view of text "B" the partitions, all of value "1" are shown as [1], and the location of the text divider is shown by [|].

At the end of the "text" there is nothing left to return:

```
↑↑----------------
{}  %fe,B/=
{}
---------------
```

The gadget which remembers where one left off in the "text" is known as the "text divider"; each text has one of its own. This divider may be moved around by the execution of a a number of different primitives or may be ordered around through the use of the "md" - "move divider" function thus:

```
↑↑----------------
{}  %md,B/=
{}
--------------
```

will return the divider to the beginning or left end of the "text" named "B".

Now to explain how to replace the partitions in a text with characters; to do this we add arguments to the expression that is used to fetch the text. For instance if we wish to fetch "B" replacing the partitions with an asterisk:

```
↑↑-------------------------------------
{}  %ft,B,*/=THE*SHACK*ON*THE*HILL
{}
------------------------------------------
```

Now we can redefine "B" as this value retuned and in other words return it to the original:

```
↑↑-------------------------
{}  %dt,B,%ft,B, //=
{}
-----------------------
```

If we now fetch "B" it would seem that the original has never been changed:

```
↑↑-------------------------------------
{}  %ft,B/=THE SHACK ON THE HILL
{}
-------------------------------------
```

Since the SAM76 language works from the inside to the outside of the expression, it first fetched "B" replacing the partitions with spaces; then on doing the next expression a "text" named "B" was defined (erasing the original and partitioned out version). This usage of interactive functions, primitives within primitives is called nesting. In theory this can be done to any depth - in other words it is limited only by the amount of memory available.

Another example of nesting:

```
{}------------------------------
{}  %dt,C,H/=
{}  %pt,B,%ft,C//=
{}  %ft,B/=TE SACK ON TF ILL
{}
------------------------------------
```

In the above example the text named "T" was partitioned on the basis of the characters received on fetching the text named "C". To return it to the original form we type:

```
{}------------------------------
{}  %dt,B,%ft,B,%ft,C///=
{}
------------------------------
```

In the latter reconstitution, text "C" was first fetched then this in the act of fetching "B" was used to replace partitions found therein; the result was then the argument of the define text expression.

At this time we will introduce a short cut in the act of "fetching". If the name of the text to be fetched is not the same as any of the primitives or built in functions then the first argument "ft" may be left out and the name of the text is used as the first argument of the expression. As we are using one character names for all our examples we can do this quite safely from now on.

In order to find out what the primitives in a system are you can do this by executing the "@f" - "what function" command thus:

```
{}------------------------------------------
{}  &@@f, /={function list will be here}
{}
------------------------------------------
```

Observe the use of the & - ampersand instead of the % sign as a warning character to start the expression; also since @ is in its own right a warning character, it is protected by preceding it with a second @, and the space after the comma is used to tell the function what you wish to use to separate the individual function mnemonics from each other.

The SAM76 language provides the ability of executing text strings and have the functions or expressions in that string executed. This is done by enclosing these executable expressions within the bounds of a "protected expression" thus inhibiting execution at the time of definition. These protected expressions are also called procedures or scripts.

```
{}------------------------------
{}  %dt,D,!%pt,B,%C////=
{}  %D/=
{}  %B/=TE SACK ON TE ILL
{}
--------------------------------
```

The fetching of "D" caused the execution of the procedure stored therein which in turn said - partition text "B" on the contents of "C".

Next we can define a text that will restore "B" to its original state:

```
{}------------------------------------
{}  %dt,E,!%dt,B,%B,%C/////=
{}  %E/=
{}  %B/=THE SHACK ON THE HILL
{}
------------------------------------
```

The part of the expression to be executed is enclosed between an ! - exclamation mark and a / - slant sign showing an executable procedure. It is easy to go from here and let the expression call another expression or itself by simply fetching the text it is contained in. This ability of recursion lets individual strings act as "subroutine" expressions.

Beginner's - Part II - Procedures

A SAM76 language procedure is a string of SAM76 functions executed when fetched by one of the three following expressions:

```
%ft,text name/ or %text name/ or &text name/
```

The latter two of the three are said to be implied fetches, that is to say that when the first argument of an expression is not the mnemonic of a resident or built in function, then a "fetch" of the "text" whose name is in the first argument is assumed.

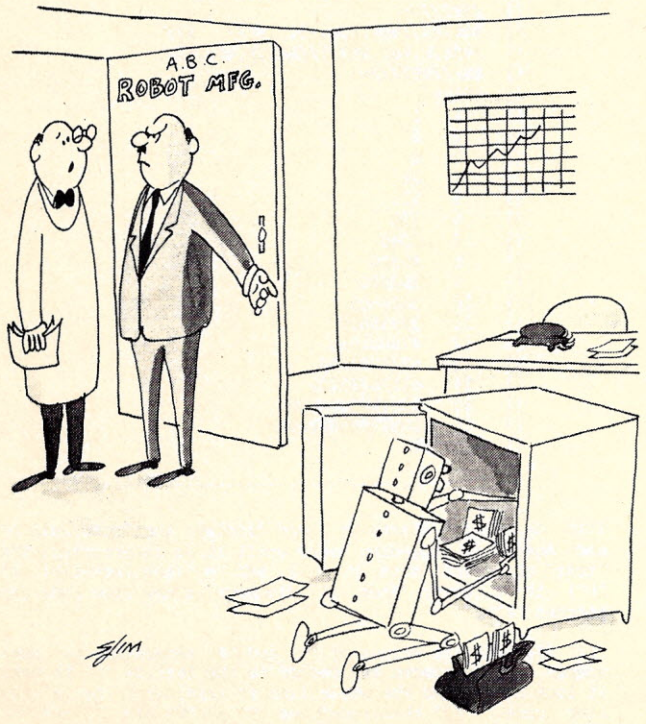
There are two other ways of protecting procedures, besides using the !...../ form; these are by using (.....) or <.....> ; in this manner you can incorporate ! and / in your text without having them act as if they were warning characters.

Let us now say that we wish to be able to fetch one string and have it partition out of "B" the contents of "C"; then output the contents of "P" and then restore "P" to its original form. To do this we need to use the "os" or "output string" primitive. This primitive outputs the contents of the second argument of its expression:

```
!!----------------------
{}  %os,ABC/=ABC
{}
~~~~~~~~~~~~~~~~~~~~
```

Now if we nest the expression that fetches "B" within the "os" expression, the contents of "B" will be displayed on execution:

```
!!---------------------------------------
{}  %os,%B//=THE SHACK ON THE HILL
{}
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```



"Oh-oh ... that's something I hadn't counted on."

We will now use "os" in an executable expresssion to display the contents of "F":

```
!!---------------------------------
{}  %dt,F,!%D/%os,%P//%D///=
{}  %F/=TE SACK ON TF HILL
{}
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

First on execution of "F", "D" was fetched. Execution of "D" caused the partitioning of "P" on the contents of "C". The execution of "os" displayed "B" as it stood with its partitions empty or "null". Then "E" was fetched and in its execution "E" caused the redefinition of "B" replacing the partitions with the contents of "C" thus restoring it back to its original condition.

Finally we would like to know just what we have created and stored in the "text area" of memory. To do this we use the "lt" or "list text" primitive; the second argument represents the character string we wish to use to precede each name just so we can tell them apart from each other thus:

```
!!---------------------------
{}  %lt, /= A C D E F P
{}
~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

In this example we used a space which precedes each name; note that "D" is last in the list - that is because it was redefined for the last time when we fetched "F" in the previous example.

Nota Bene The editor of this beginner's description of the SAM76 language wishes to credit Robert M. Evans, from whose first technical writing effort this was derived.

Note that in the definition of procedures it is necessary to protect the functions to prevent immediate execution; for example:

```
!!-----------------------------------------
{}  %dt,A,!%os,THIS IS A PROCEDURE///=
{}
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

If there are partitions in the procedure they will be replaced in the same manner as when fetching ordinary strings of text with partitions; thus arguments can be plugged in at the time of execution:

```
!!---------------------------------
{}  %dt,SQUARE,!%MU,*,*///=
{}  %pt,SQUARE,*/=
{}  %SQUARE,12/=144
{}
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

nested functions

Functions can be nested to eliminate the need for storing a text in memory if it is to be nested only once; for instance:

```
%os,WHAT IS YOUR NAME?- /%os,
WELL HELLO THERE %is//
```

The above procedure will display "WHAT IS YOUR NAME?- " then it will input a string from the keyboard and display "WELL HELLO THERE " followed by the string read in from the keyboard.

It will input the string first because the input string function is nested within the output string expression so the input string expression will have to be evaluated to make the output string function complete with a value.

Another example of nested functions is concatenating (joining end to end) something onto a string:

```
!!----------------------------------
{}  %dt,A,&ft,A/ SOME TEXT/=
{}
----------------------------------
```

This will fetch "A" and place the contents where the fetching expression was. Then it will redefine "A" as the previous value plus whatever else was put in the define text expression.

When expressions are nested the innermost expression is evaluated first and the value is placed where the expression used to be.

To make a procedure loop, all that is necessary is to place a fetching expression to the procedure within the procedure thus:

looping and recursion

```
!!-----------------------------------------
{}  %dt,PROC,!%os,
{}  THIS PROCEDURE LOOPS/%PROC///=
{}  %PROC/=
{}  THIS PROCEDURE LOOPS
{}  THIS PROCEDURE LOOPS
{}  THIS PROCEDURE LOOPS
{}  THIS PROCEDURE LOOPS
{}  THIS PROCEDURE LOOPS
{}  THIS PROCEDURE LOOPS
{}  <sce-pro>
{}
-----------------------------------------
```

The above example would keep running indefinitely, and so a means of emergency interruption is provided through the operation of the "rub out" or "delete" key during the procedure execution; termination in this manner is indicated by the "special condition exit" message. If the looping procedure had partitions in it, these would be replaced during the fetching operation.

Short recursive procedures can be written to do such things as factorial, square roots or exponentiation. Following is an example of a procedure to take the factorial of a number:

```
!!-------------------------------------------------
{}  %dt,F,
{}  !%ii,*,1,1,!%mu,*,%F,%su,*,1//////////=
{}  %pt,F,*/=
{}  %F,5,/=120
{}
-------------------------------------------------
```

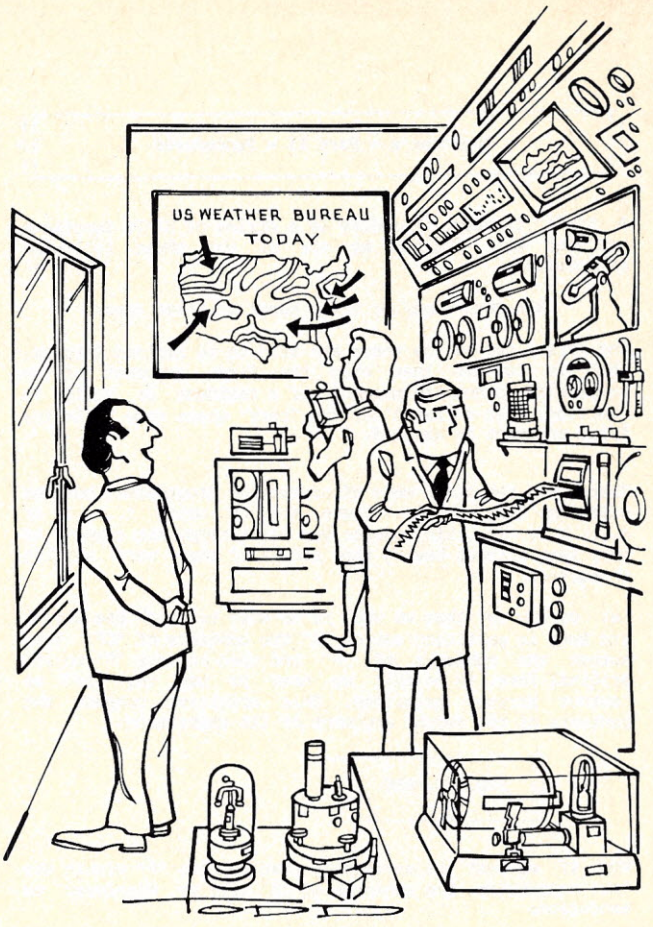
Note the extra number of slant signs used in the expression that defines "F" is not really required, but it is safer to put a few extra / signs than too few, and the count can be a little hairy at times.

In the foregoing example the procedure first tests to see whether or not the number (in the partition which replaced the asterisk) is one; if it is then the factorial is one, else the factorial is the number times the factorial of the number minus one which is computed by fetching again "F".

Another example of recursive procedure is the following expression to do exponentiation:

```
!!-------------------------------------------------
{}  %dt,POWER,!%ii,EXP,1,BASE,
{}  !%mu,BASE,%POWER,BASE,%su,EXP,1///////=
{}  %pt,POWER,BASE,EXP/=
{}  %POWER,2,4/=16
{}
-------------------------------------------------
```

This procedure is quite similar to the one for factorial; first it tests to see whether or not the exponent (partition [2] or EXP) is equal to one. If it is so equal, then the result is the BASE (partition [1]), else the result is the base times the base raised to the power of the exponent minus one, which is computed by recursion, as before. In the end all the multiplications are performed and the result is left.



"Looks like it might be a nice day tomorrow!"

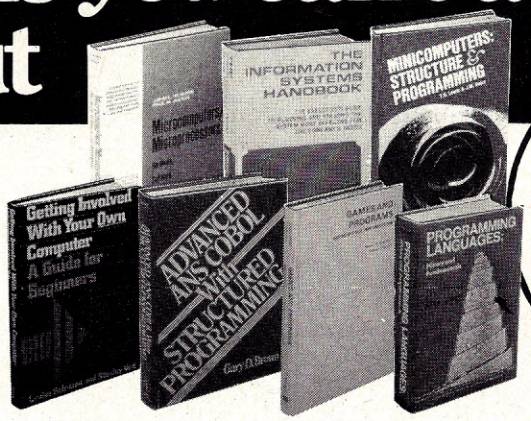
The following is an example of a procedure that is not recursive:

```
!!-----------------------------------------
{}  %dt,F1,!%dt,N,1/%dt,X,1/%os,
{}  /%F2///=
{}  %dt,F2,!%os,%ps,-3, ,%N//   %F/
{}  /%dt,N,%ad,%N/,1//%dt,X,%mu,%N/,
{}  %X///%F2///=
{}  %F1/=
{}    1   1
{}    2   2
{}    3   6
{}    4   24
{}    5   120
{}    6   720
{}    7   5040
{}    8   40320
{}    9   362880
{}   10   3628800
{}   11   39916800
{}   12   479001600
{}   13   6227020800
{}   14   87178291200
{}   15   1307675368000
{}   16   20922789888000
{}  <sce-F2>
{}
-----------------------------------------
```

This procedure starts at 1 and displays the factorials of each number in ascending order until it is interrupted. The "text" named "F1" sets "N" to 1, and "X" (the factorial of "N") to 1 also; then it displays a new line code and fetches "F2".

"F2" displays the values of "N" and "X" separated by some spaces - some spaces are padded to the left of "N" to make it look nice - and the whole line is terminated by a new line code. "F2" then redefines "N" as "N" plus 1, and "X" as the value of "N" times the value of "X". Lastly "F2" fetches itself again.

EDP books you can't afford to be without



any one of these great professional books for only **$1.89** values up to $35.00

MICROPROCESSOR APPLICATIONS MANUAL
by Motorola, Inc.
435/278 Pub. Pr., $28.50 Club Pr., $22.50

HANDBOOK OF CIRCUIT ANALYSIS LANGUAGES AND TECHNIQUES
by R. Jensen & L. McNamee
769/656 Pub. Pr., $34.50 Club Pr., $28.50

PROGRAMMING LANGUAGES: History & Fundamentals
by J. E. Sammett
767/513 Pub. Pr., $25.00 Club Pr., $19.95

GAMES & PROGRAMS MATHEMATICS FOR MODELING
by R. R. Singleton & W. Tyndall
767/815 Pub. Pr., $13.00 Club Pr., $10.75

GETTING INVOLVED WITH YOUR OWN COMPUTER— A Guide For Beginners
by L. Solomon & S. Veit
771/952 Pub. Pr., $9.95 Club Pr., $8.35

ILLUSTRATING BASIC (A Simple Programming Language)
by D. Alcock
771/928 Pub. Pr., $10.95 Club Pr., $8.95

ADVANCED ANS COBOL WITH STRUCTURED PROGRAMMING
by G. D. Brown
772/118 Pub. Pr., $19.95 Club Pr., $15.95

GRAMMARS FOR PROGRAMMING LANGUAGES
by J. C. Cleaveland & R. C. Uzgalis
783/594 Pub. Pr., $15.95 Club Pr., $12.75

MINICOMPUTER SYSTEMS Organization and Programming
by R. H. Eckhouse, Jr.
768/641 Pub. Pr., $18.95 Club Pr., $13.95

THE COMPULATOR BOOK— Building Super Calculators & Minicomputer Hardware with Calculator Chips
by R. P. Haviland
783/578 Pub. Pr., $10.95 Club Pr., $9.30

BUILD YOUR OWN WORKING ROBOT
by D. Heiserman
769/311 Pub. Pr., $8.95 Club Pr., $7.60

MICROCOMPUTERS/MICROPROCESSORS: Hardware, Software and Applications
by J. L. Hilburn & P. N. Julich
771/499 Pub. Pr., $19.50 Club Pr., $15.60

MINICOMPUTERS: Structures and Programming
by T. G. Lewis & J. W. Doerr
773/009 Pub. Pr., $13.95 Club Pr., $10.95

THE INFORMATION SYSTEMS HANDBOOK
by F. W. McFarlan & R. L. Nolan
769/29X Pub. Pr., $35.00 Club Pr., $25.50

MICROPROCESSOR PROGRAMMING FOR COMPUTER HOBBYISTS
by N. Graham
783/56X Pub. Pr., $12.95 Club Pr., $10.95

LET'S TALK LISP
by L. Siklossy
770/077 Pub. Pr., $13.95 Club Pr., $10.95

Introductory offer to new members of the Computer Professionals' Book Club

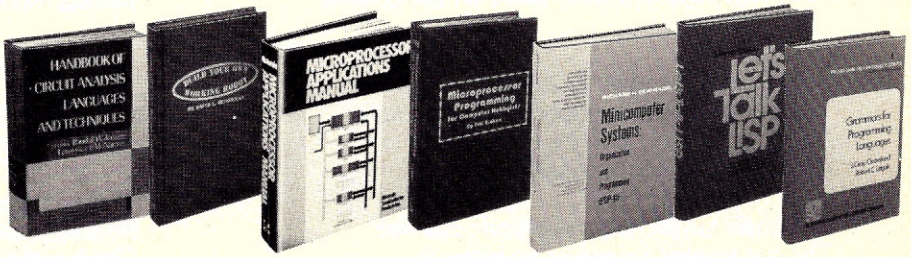
Special $1.89 bonus book comes to you with your first club selection

THIS new professional club is designed to meet your day-to-day on-the-job needs by providing practical books in your field on a regular basis at *below publisher prices.* If you're missing out on important technical literature—if today's high cost of reading curbs the growth of your library—here's the solution to your problem.

The Computer Professionals' Book Club was organized for you, to provide an economical reading program that cannot fail to be of value. Administered by the McGraw-Hill Book Company, all books are chosen by qualified editors and consultants. Their understanding of the standards and values of the literature in your field guarantees the appropriateness of the selections.

How the Club operates: Every month you receive free of charge *The Computer Professionals' Book Club Bulletin.* This announces and describes the Club's featured book of the month as well as alternate selections available at special members' prices. If you want to examine the Club's feature of the month, you do nothing. If you prefer one of the alternate selections—or if you want no book at all—you notify the club by returning the card enclosed with each *Bulletin.*

As a Club Member, you agree only to the purchase of four books (including your first selection) over a two-year period. Considering the many books published annually, there will surely be at least four you would want to own anyway. By joining the club, you save both money and the trouble of searching for the best books.



VALUES UP TO $35.00 WITH MAJOR DISCOUNTS ON ALL OTHER CLUB SELECTIONS. Your bonus books come with the first selection, and you may choose both of them from the books described in this special introductory offer.

EXTRA SAVINGS: Remit in full with your order, plus any local and state tax, and McGraw-Hill will pay all regular postage and handling charges.

NO RISK GUARANTEE:
If not completely satisfied return selections for full refund and membership cancellation.

MAIL THIS COUPON TODAY

COMPUTER PROFESSIONALS/Book Club P.O. Box 582 Princeton Road, Hightstown, New Jersey 08520

Please enroll me as a member and send me the two books indicated. I am to receive the bonus book at the introductory price of $1.89 plus my first selection, plus tax, postage, and handling. If not completely satisfied, I may return the books within 10 days and request that my membership be cancelled. If I keep the books, I agree to take a minimum of three additional books during the next two years at special Club prices (guaranteed 15% discount, often more). I will receive the Club Bulletin 13 times a year. If I want to examine the featured selection, I need take no action. It will be shipped automatically. If, however, I want an alternate selection—or no book at all—I simply notify the Club by returning the convenient card always enclosed. I will always have a minimum of 10 days in which to return the card and you will credit my account fully, including postage, if this is not the case. Membership in the club is continuous but cancellable by me at any time after the four-book purchase requirement has been filled. *This order subject to acceptance by McGraw-Hill. Orders from outside the continental U.S. must be prepaid. Company, business, or institutional tax exemption status is not appliable to purchases made through individual Club memberships. All prices subject to change without notice. Offer good for new members only.*

Write Code # of $1.89 bonus book selection here

Write Code # of first selection here

NAME _____

ADDRESS _____

CITY _____

STATE _____ ZIP _____

P39301

CIRCLE 110 ON READER SERVICE CARD

(These are Marey's photos)

1. The cat acquires a velocity
At thirty-two feet per second per second
And everything begins to blur — the tail
Is twirling, the cat is turning, the paws

Are on the ground. Again the cat has turned
From upside-down around to downside-down
Without a wall to push on or a string
To pull itself around on the way down.

What is the mechanism? What is the
Solution to the problem of the cat,
Released from rest and oriented up,
Descending in a circle in a line

Of gravity, the quickest thinker, down,
To always land with four paws on the ground?

2. In *Comptes Rendus,* in 1894,
The cat stop falls in photographs Monsieur
Marey has taken with his camera.
His explanation is a hopeless use

Of words for pictures: everyone can see
The cat superimposed upon himself:
The first slow feet, and then the faster feet
Prepared to meet the sidewalk half way down.

Monsieur Marey contorts his vertebrae
With words like *torsion, opposition, tors*
And wraps his tail around his helix spine
Once clockwise for a counterclockwise half

Rotation of the animal. We see
Two human hands still grasping for the cat.

3. Herr Magnus (no computer) had to crank
The torso of the cat manually
Through its manoeuvres. So he simplified
His model for the numbers it would use.

Dr. McDonald, physiologist,
Dealt with the phenomenological
Aspects. His cat was not an equation
That did its business neatly on the paper.

Professors Kane and Scher were fortunate.
A man on the moon supported them
And NASA sectioned cat cadavers so
The moments of inertia of the cat

That Kane and Scher did use were accurate
Enough to lend credence to their results.

4. A couple of cylinders is their cat,
Without a head, with negligible legs,
Without a tail: a mechanical Manx.

This cat possesses a Lagrangian,
Potential and kinetic energy
Confused in an expression for the cat

Falling, the cat jumping, the cat at rest.
The lithe Lagrangian, ready to be
The cat in the clutches of gravity

Submits to differentiation with
Respect to time and with respect to speed
To fall in a falling and revolving mode.

Released now, the equations, upside down,
Descend in the computer, and they turn.

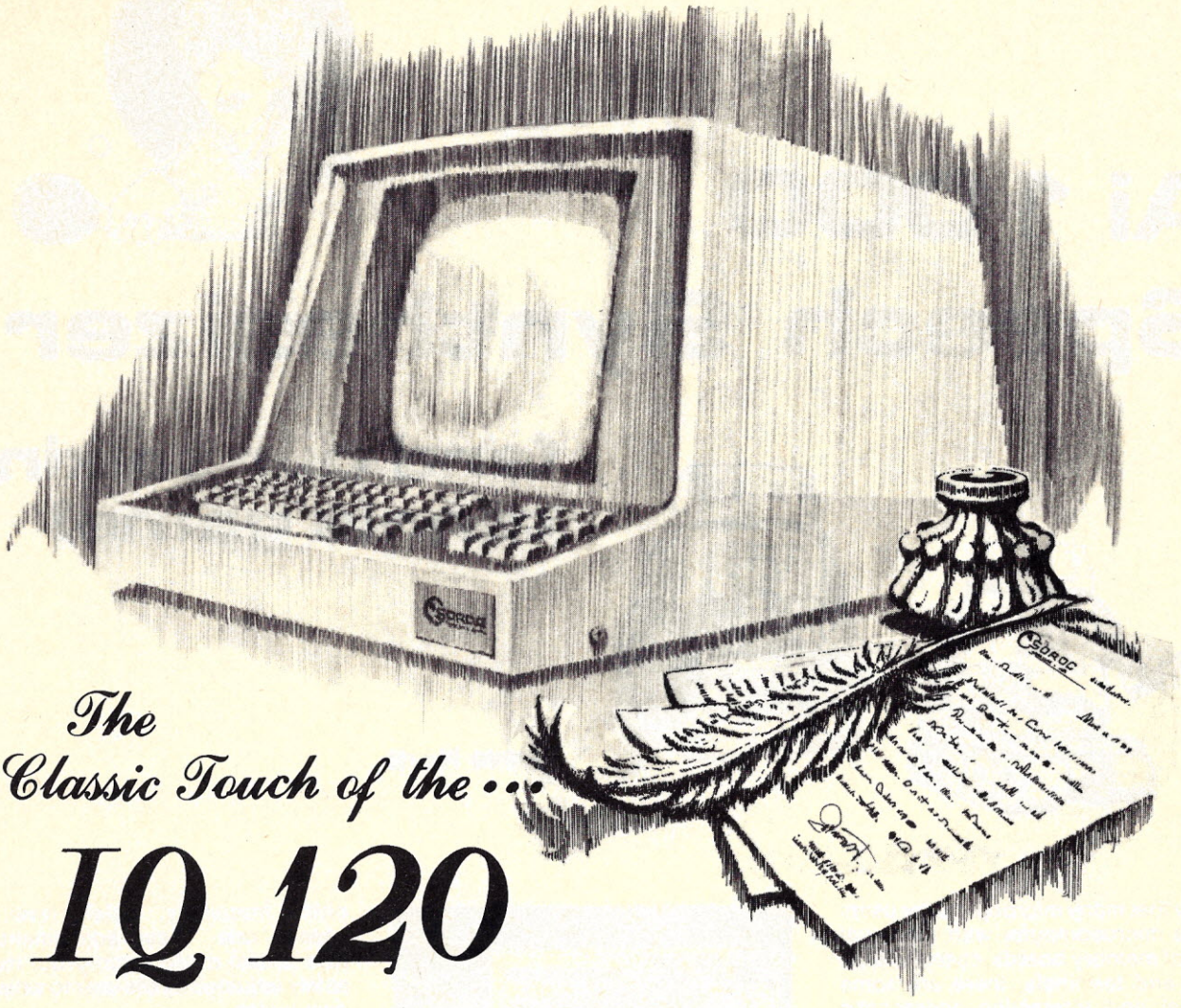
References.

M. Marey, Des mouvements que certains animaux exécutent pour retomber sur leurs pieds, lorsqu'ils sont précipites d'un lieu élevé. *C. r. hebd. Séanc. Acad. Sci. Paris 119,* 714-717 (1894).

R. Magnus, Wie sich die fallende Katze in der Luft umdreht. *Archs. Neerl. Physiol. 7,* 218-222 (1922).

D. A. McDonald, How *does* a falling cat turn over? *St. Bart's Hosp. J. 56,* 254-258 (1955).

T. R. Kane and M. P. Scher, A dynamical explanation of the falling cat phenomenon, *Int. J. Solids Structures 5,* 663-670 (1969).



The Classic Touch of the • • •

IQ 120

Clearly displays the results of a careful blending of Old World Elegance with today's New World of Technology. This blending is what creates character in the IQ 120, and such refined features as: upper and lower case / 24 line × 80 characters / address cursor / tabbing / dual intensity / numeric pads / auto repeat / 15 baud rates / aux. port and optional printer port.

so if you are one who is serious about what terminal to buy, contact SOROC where character along with quality and design excellence is part of each and every display • • • • • • • • $995.00

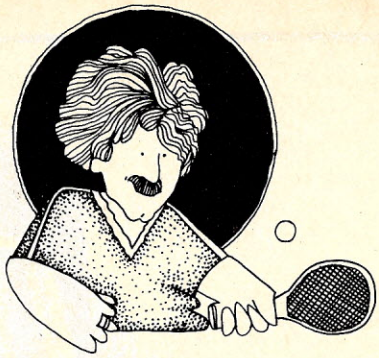
SOROC TECHNOLOGY, INC.

165 FREEDOM AVE., ANAHEIM, CA 92801

714·992-2860 • 800·854-0147

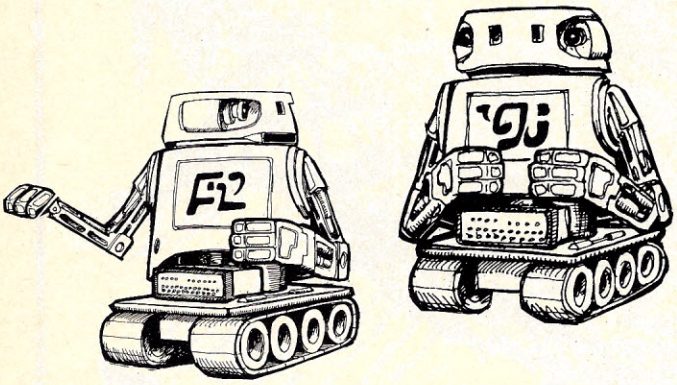
CIRCLE 164 ON READER SERVICE CARD

A Creative Computing Equipment Profile. . .



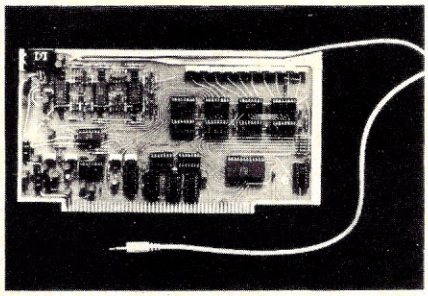
Ai 1000 Speech Synthesizer

(and our talking tennis game)



Steve North

Now that many microcomputer users have a microcomputer with the usual array of memory boards, cassette-tape units, and terminals, there are some unusual and interesting accessories appearing on the market, formerly available only on large-scale computers at large-scale prices. Among these: music synthesis boards, color-graphics interfaces, boards for controlling external devices, speech synthesis boards, and speech-recognition units. In this review we'll take a closer look at the Ai Cybernetic Systems Model 1000 Speech Synthesizer System for S-100 bus computers. The Model 1000 Speech Synthesizer costs around $380, and requires only a connection to an external audio amplifier to be used.



Ai Cybernetic Systems Model 1000 Speech Synthesizer is a hardwired analog of the human vocal tract; the cord connects to an external audio amplifier.

The Model 1000 Speech Synthesizer hardware itself consists of an analog model of the human vocal tract, and digital logic to interface this to the computer. The analog circuitry simulates two basic types of sounds used in the English language: voiced sounds made by the larynx, and non-voiced sounds (made by rushing air). The sounds are passed through an array of ten active filters that "simulate the formant frequencies associated with the preferred energy passage of the resonant cavities of the mouth, nose, tongue, and teeth." The summed output of the filters is then spectrally compensated so it will match the properties of the human voice. The digital logic on the board decides when the board is being addressed, and controls the analog portion of the speech synthesis unit. There are quite a few trim pots on the board, for adjusting various portions of the analog model. Ten of these are used to adjust the active filters. This is done at the factory, and the user is advised not to touch these at all. There are four more controls, used for: noise level, pitch frequency, voice level, and speech rate. These adjustments are also preset at the factory but may be safely adjusted by the user to satisfy his own tastes.

The Model 1000 hardware is very simple to program. The unit uses one I/O port, located at 254 decimal. The board is programmed to speak as easily as one programs a printer to print. A character output to the port is spoken as a predetermined corresponding "phoneme" (a unit of speech, a single sound). One bit of the input port tells the processor when the unit is ready to speak another phoneme. Thus to say a word, one has only to decide what sounds are contained in it, convert these to certain characters, then output these characters to the speech synthesizer unit.

The character symbols that represent phonemes were chosen to suggest the actual sounds they make. For example, "A" is spoken as the A in STAY. "E" is spoken as E in ZEBRA, and "M" as the M in AM. However, there are some symbols that do not resemble the sounds made when they are spoken by the Model 1000. For example, "+" means a *th* sound, as in THAW. Likewise, the "#" is used to make an *er* sound, as in bird or computer. A few sounds are programmed by two sym-

English: "I AM A TALKING ROBOT"
Program: &&IE AM AE T)..KEN RO.B)..T

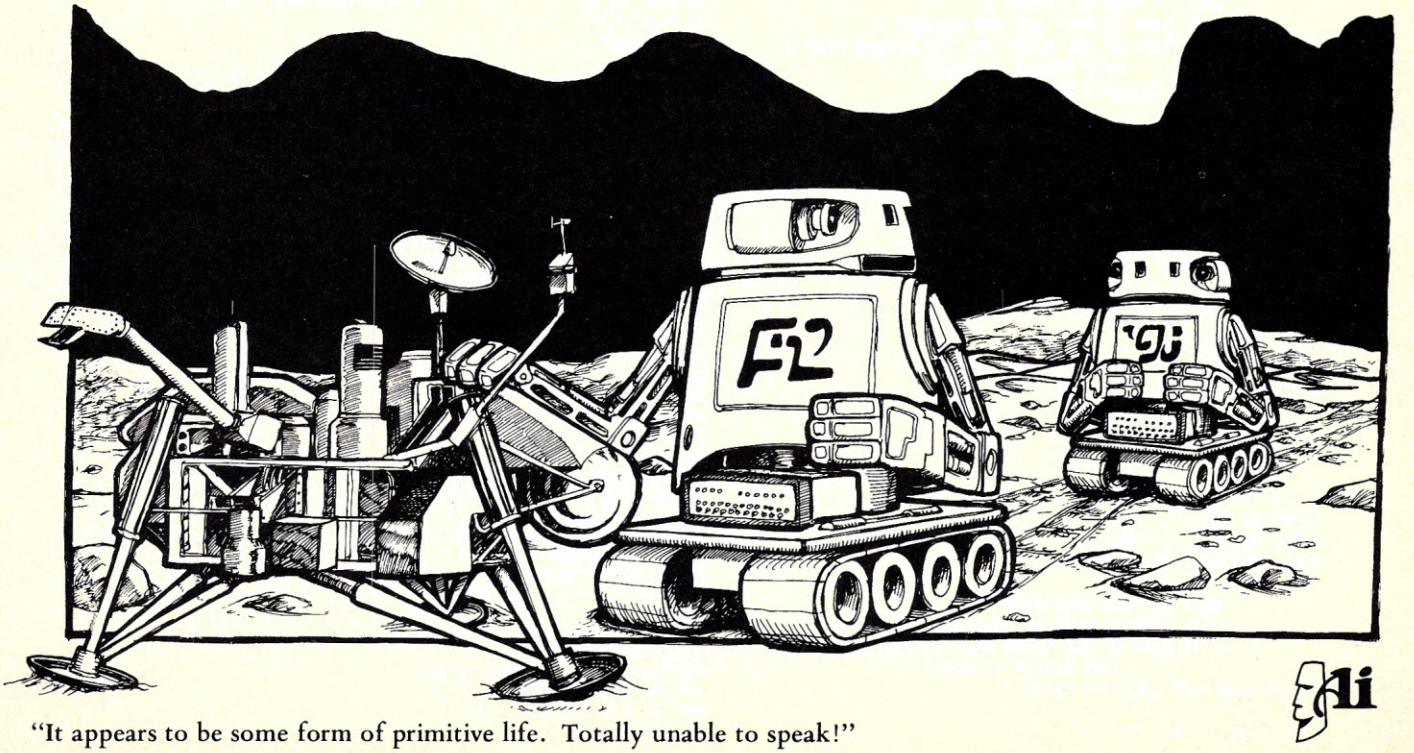
bols such as the CH phoneme (as in CHINA), which is represented by "TC".

The manual contains a very good explanation (for a novice like myself, at least) of how the spoken language is broken up into different sounds, and how these can then be encoded for the Model 1000. In general, the characters used to represent sounds are as similar as possible to the written characters. (For instance, IIZZ is used to represent the word IS). However, because there are many more sounds in the language than characters, some special characters (&, /, ', etc.) are also used. So in the Model 1000 Speech Synthesis System's language, /(!!.T really means SHOULD. The manual does explain in great detail what characters should be used to make what sounds, and includes rules used for conversion. A helpful glossary is included.

Although the speech synthesizer may be programmed directly at an assembly-language level, you'll probably prefer to program it in BASIC. This would allow you to have speaking computer games! The Model 1000 manual includes three speaking BASIC programs: Lunar Lander, a version of HELLO (the computer introduces itself), and a very short program used for testing your own words and sentences for speech synthesis. A machine-language subroutine actually outputs the characters to the board. Although the board could have been controlled directly from BASIC, using INP and OUT statements, it is very important that the board be programmed with a new phoneme as soon as possible after it has finished a previous one, to improve the quality of the speech. So a machine-language subroutine, called from BASIC (which supplies the phonemes to be spoken) is used. At the beginning of a program, the machine-language program is POKED into a predetermined area of memory. Then, when a certain word is to be spoken, it is placed in a character-string variable and a BASIC subroutine is called. This subroutine then calls the machine-language subroutine, speaking the word one phoneme at a time, and returns to the part of the BASIC program which wanted something spoken in the first place.

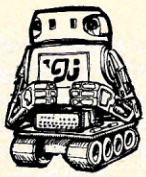
The crucial question in all this is, how does it sound? The truth is, one must bend his ear a bit to understand what the Model 1000 is trying to say. I was able to understand most of the words in the glossary provided in the manual, as well as those in the sample program (see below). However, it's a great advantage to know in advance what the machine might be trying to say. Steve Gray, our Editor-in-Chief, who didn't know what the unit was trying to say, commented that he didn't understand a single word it was saying. I thought that its pronunciation of words such as "ENTERPRISE" ("N .. T### .. P. R&&IEZZ) and ASCII (&&ZZ ... KEE) was quite good, while words like "ROBOT" (460U .. B) ... T) and "CYBERNETIC" (SS&IE.B## .. N".. TII ... K) were rather weak.

If you want to experiment with computer-speech synthesis, the Model 1000 would be worth investigating. It isn't necessary to buy one without hearing it first — you can check the capabilities of speech synthesis units out at a computer festival. Ai Cybernetic Systems also sells a demonstration tape on audio cassette of the Model 1000 for $5, and the manual for $4 (both are $7.50).



"It appears to be some form of primitive life. Totally unable to speak!"

TENNIS MATCH



A *talking* game!

Steve North

```
     ARE YOU READY...HERE WE GO!!!


SERVE!   TYPE? S
     SERVE IS GOOD...CAN'T BE RETURNED!

          SCORE   15 - LOVE


SERVE!   TYPE? S
     SERVE IS GOOD...CAN'T BE RETURNED!

          SCORE   30 - LOVE


SERVE!   TYPE? S
     SERVE IS GOOD...CAN'T BE RETURNED!

          SCORE   40 - LOVE


SERVE!   TYPE? L
     SERVE IS BAD
     SERVE AGAIN!     TYPE? L
     SERVE IS BAD...DOUBLE FAULT!

          SCORE   40 - 15


SERVE!   TYPE? L
     SERVE HAS BEEN RETURNED...

               WHAT IS YOUR POSITION? 3
               WHAT TYPE OF SHOT ARE YOU MAKING? S
               WHAT PART OF THE COURT ARE YOU AIMING FOR? 2
                    YOUR RETURN IS GOOD!
  NICE SHOT-THE COMPUTER COULDN'T REACH IT

          SCORE   GAME
--------------- GAME OVER ---------------
          SCORE-GAMES      YOU...ME
                            1      0



SERVE!   TYPE? L
     LET SERVE...TAKE 2
SERVE!   TYPE? S
     SERVE IS BAD
     SERVE AGAIN!     TYPE? S
     SERVE IS BAD...DOUBLE FAULT!

          SCORE   LOVE - 15


SERVE!   TYPE? L
     SERVE IS BAD
     SERVE AGAIN!     TYPE? L
     SERVE HAS BEEN RETURNED...

               WHAT IS YOUR POSITION? 1
               WHAT TYPE OF SHOT ARE YOU MAKING? S
               WHAT PART OF THE COURT ARE YOU AIMING FOR? 3
                    YOUR RETURN IS GOOD!
  NICE SHOT-THE COMPUTER COULDN'T REACH IT

          SCORE   15 - 15
```

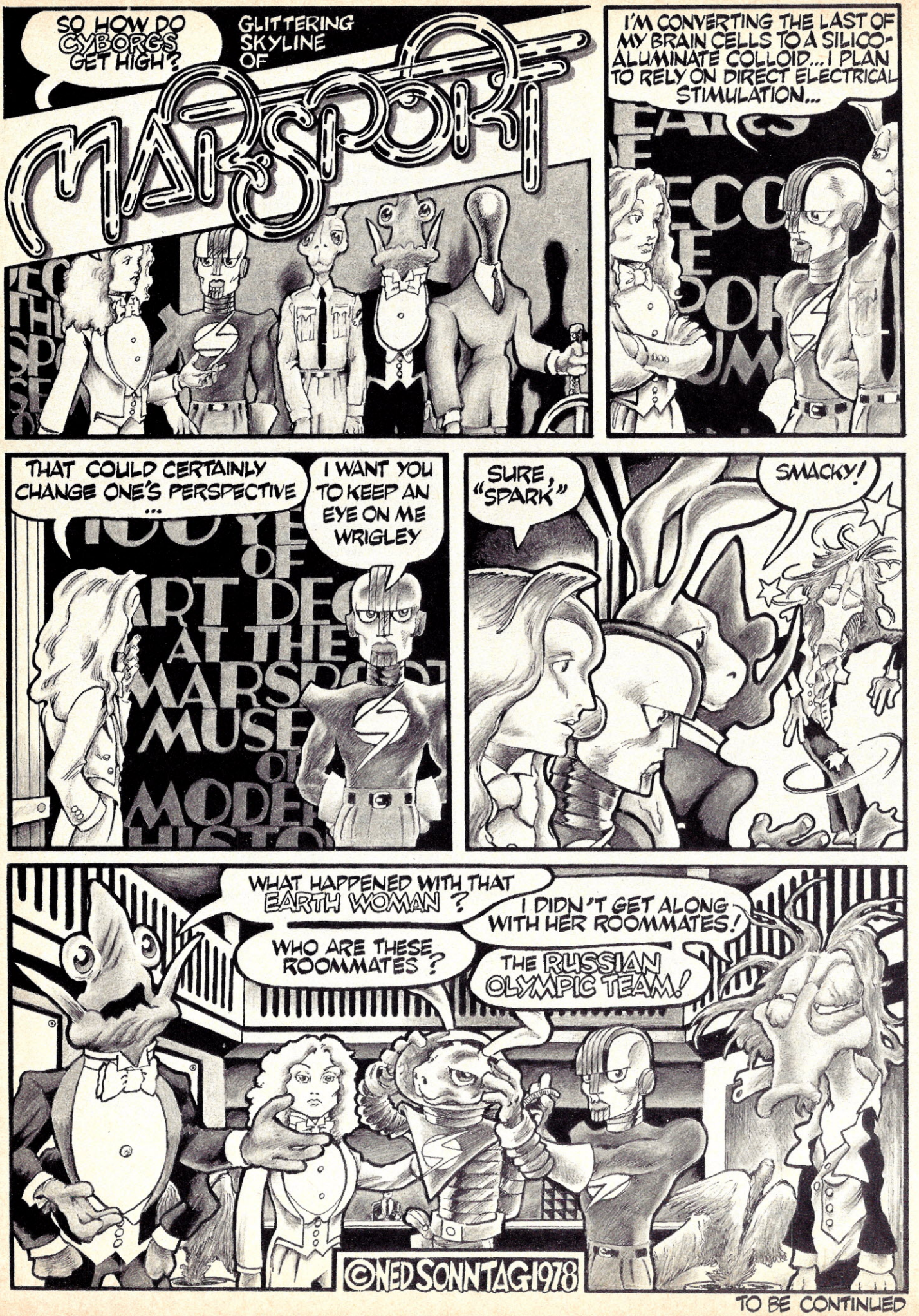
To show you how the Model 1000 is programmed for speech we've written a speaking tennis game. The original game was written by V. Nahigan and Dave Ahl, with modifications for speech synthesis by the writer. The part of the program that loads in the machine-language subroutine is contained in lines 10-66. The routine is loaded starting at location 12201 decimal, and is designed to be run in a 12K computer with MITS (Microsoft) 8080 BASIC. (The program is a rather tight fit, so the instructions were removed). When we want the speech synthesis unit to say something, the string of phonemes is placed in V$, and W$ is used for pitch control. 1's are spoken normally, 2's are stressed. This helps to make the speech a bit more understandable. Then a GOSUB 5000 causes this to be spoken. When a phoneme is spoken and the unit is ready for another, it continues to voice the previous one until reprogrammed (on and on and...) so line 5110 must be used to silence the unit when it has finished the word or sentence. In this program, only certain phrases are pronounced as the game proceeds, while the scores, for instance, aren't read off.

Since the instructions were removed from the program, here they are. Shots are designated by S or L, for Slam or Lob. Areas of the court are referred to by number, 1, 2, 3, or 4, which refer to left backcourt, right backcourt, left forecourt, and right forecourt. On serves (you always serve first), you only input the type of shot, S or L. ■

Program Listing

```
5 PRINT "TENNIS MATCH"
10 REM     SPEECH SYNTHESIS INITIALIZATION CODE
20 WT=128
22 SF=64
24 ST=63
30 POKE 73,169
35 POKE 74,47
40 SA=12201
42 LN=28
44 DATA 33,177,47,229,42,4,0,233,219,254
46 DATA 230,1,202,177,47,230,0,198
48 DATA 20,214
50 DATA 1,194,188,47,123,211,254,201
60 FOR II=SA TO SA+LN-1
62 READ WD
64 POKE II,WD
66 NEXT II
100 PRINT
320 PRINT TAB(10);"ARE YOU READY...HERE WE GO!!!"
325 REM  NOW THE COMPUTER SAYS, 'HERE WE GO'
330 V$="HEE##  WEE  GOO"
340 W$="121111121111211"
341 GOSUB 5000
350 Y=0
360 Z=0
370 PRINT
380 PRINT
390 PRINT
400 PRINT "     SERVE!   TYPE";
404 V$="SS####V         T&IEE.P"
406 W$="11211111111111111121111"
408 GOSUB 5000
410 INPUT A$: IF A$<>"L" AND A$<>"S" THEN PRINT "'L' OR 'S'": GOTO 400
420 A=100*RND(1)
430 IF A$="L" THEN 520
440 C=6
450 D=51
460 IF A<C THEN 500
470 IF A<D THEN 700
480 PRINT TAB(10);"SERVE IS BAD"
482 V$="SS####V  IIZZ   B'&&'.T"
484 W$="112111111211111211111"
486 GOSUB 5000
```

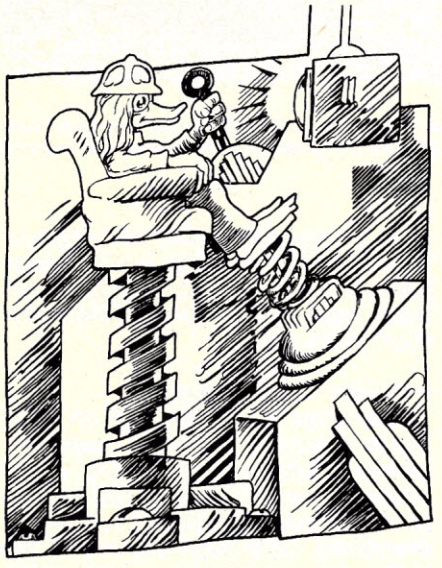
```
490 GOTO 550
500 PRINT TAB(10);"LET SERVE...TAKE 2"
502 V$="L'''.T   SS###V      T.AE...K  T.OUU"
504 W$="2211111111221111111111122111111111211"
506 GOSUB 5000
510 GOTO 400
520 C=4
530 D=66
540 GOTO 460
550 PRINT TAB(10);"SERVE AGAIN!     TYPE";
552 V$="SS###V  II6'''IIN       T&IEE.P"
554 W$="11211111121121111111111111121111"
556 GOSUB 5000
560 INPUT B$: IF B$<>"L" AND B$<>"S" THEN PRINT "'L' OR 'S'": GOTO 550
570 E=100*RND(1)
580 IF B$="L" THEN 670
590 G=5
600 H=41
610 IF E<G THEN 650
620 IF E<H THEN 700
630 PRINT TAB(10);"SERVE IS BAD...DOUBLE FAULT!"
632 V$="SS###V  IIZZ  B'&&'.T  D&!!..BLLL  F.&&&L.T"
634 W$="112111112211111121111112211111111111111"
636 GOSUB 5000
640 GOTO 1300
650 PRINT TAB(10);"LET SERVE...TAKE 1"
652 V$="L'''.T  SS###V     T.AE...K  W!!!N"
654 W$="2211111112211111111221111121111"
656 GOSUB 5000
660 GOTO 550
670 G=3
680 H=76
690 GOTO 610
700 I=100*RND(1)
710 IF I>6 THEN 740
720 PRINT TAB(10);"SERVE IS GOOD...ACE!"
722 V$="SS###V  IIZZ  G'!!.T    AESSS"
724 W$="11211111221111112111111122111"
726 GOSUB 5000
730 GOTO 1280
740 K=100*RND(1)
750 IF A$="L" THEN 810
760 IF B$="L" THEN 810
770 N=61
780 IF K<N THEN 850
790 PRINT TAB(10);"SERVE IS GOOD...CAN'T BE RETURNED!"
792 V$="SS###V  IIZZ  G'!!.T  K.''IINN..T  B'EE  RE.T##.D"
794 W$="1121111122111111221111112211111111112111121111111"
796 GOSUB 5000
800 GOTO 1280
810 N=76
820 GOTO 780
850 PRINT TAB(10);"SERVE HAS BEEN RETURNED..."
852 V$="SS###V  H&&ZZ  B'''INN  R.T##.D"
854 W$="11211111122111112111111111111"
856 GOSUB 5000
860 PRINT
870 O=INT(4*RND(1))+1
880 PRINT TAB(20);"WHAT IS YOUR POSITION";
882 V$="W&&&.T  IIZZ  90###  P.ZII'/C&&&.N"
884 W$="12111111221111221111111121111111"
886 GOSUB 5000
890 INPUT Q
900 IF O+Q=5 THEN 1240
910 PRINT TAB(20);"WHAT TYPE OF SHOT ARE YOU MAKING";
912 V$="W&&&.T  T&IEE.P  &&V  /C(..T"
914 W$="12111111121111111111111111"
916 GOSUB 5000
920 INPUT C$
930 PRINT TAB(20);"WHAT PART OF THE COURT ARE YOU AIMING FOR";
932 V$="W&&&.T  P(###.T  &&V  T.&!!  K###.T"
934 W$="12111111112111111211111211111121111"
936 GOSUB 5000
940 INPUT R
950 S=100*RND(1)
960 IF C$="L" THEN 990
970 IF S<81 THEN 1070
980 GOTO 1000
990 IF S<91 THEN 1070
1000 U=4*RND(1)
1010 PRINT TAB(30);"YOUR RETURN IS BAD..."
1012 V$="REET.####N  IIZZ  B'&&'.T"
1014 W$="11111222111211111112111"
1016 GOSUB 5000
1020 IF U<2 THEN 1050
1030 PRINT TAB(33);"HIT OUT-OF-BOUNDS"
1040 GOTO 1300
1050 PRINT TAB(33);"HIT INTO NET"
1060 GOTO 1300
1070 PRINT TAB(30);"YOUR RETURN IS GOOD!"
1072 V$="90###  RE.T###.N  IIZZ  G'!!!.T"
1074 W$="11211112211211111122111111121111"
1076 GOSUB 5000
1080 A1=INT(4*RND(1))
1090 IF R+A1=5 THEN 1270
1100 W=100*RND(1)
1110 REM
1120 IF W<84 THEN 1220
1130 GOTO 1150
1140 IF W<84 THEN 1220
1150 C1=4*RND(1)
1160 PRINT TAB(30);"COMPUTER'S RETURN IS BAD..."
1162 V$="&IEE  MII/C.8T  II..T"
1164 W$="1221111211111111121111"
1166 GOSUB 5000
1170 IF B<2 THEN 1200
1180 PRINT TAB(33);"HIT OUT-OF-BOUNDS"
1190 GOTO 1280
1200 PRINT TAB(33);"HIT INTO NET"
1210 GOTO 1280
1220 PRINT TAB(30);"COMPUTER'S RETURN IS GOOD!"
1222 V$="M&IEE /C(..T  W!!ZZ  G'!!.T"
1224 W$="12211111111111121111112111"
1226 GOSUB 5000
1230 GOTO 860
1240 PRINT "   NICE TRY-YOU WERE UNABLE TO REACH THAT SHOT-COURT #"O
1242 V$="90UU  K.(!!.T.NT  46EE/C  II..T"
1244 W$="2211111111111111111121111221111"
1246 GOSUB 5000
1250 GOTO 1300
1270 PRINT "   NICE SHOT-THE COMPUTER COULDN'T REACH IT"
1272 V$="&IEE K.(!!.T.N.T 46EEE/C  II..T"
1274 W$="12111122211111111112111111111111"
1276 GOSUB 5000
1280 Y=Y+1
1290 GOTO 1310
1300 Z=Z+1
1310 PRINT: GOSUB 2000
1320 PRINT TAB(15);"SCORE   "S$
1340 IF Y>=4 AND Y>Z+1 THEN 1370
1350 IF Z>=4 AND Z>Y+1 THEN 1390
1360 GOTO 380
1370 Y1=Y1+1
1380 GOTO 1400
1390 Z1=Z1+1
1400 PRINT "--------------- GAME OVER ---------------"
1410 PRINT TAB(15);"SCORE-GAMES      YOU...ME"
1420 PRINT TAB(32);Y1"     "Z1
1430 IF Y1>=6 AND Y1>Z1+1 THEN 1460
1440 IF Z1>=6 AND Z1>Y1+1 THEN 1490
1450 GOTO 350
1460 PRINT
1470 PRINT "*****CONGRATULATIONS...YOU WON*****"
1472 V$="90UUU W!!N"
1474 W$="1221111111"
1476 GOSUB 5000
1480 GOTO 1510
1490 PRINT
1500 PRINT "*****AS PREDICTED, THE COMPUTER IS AGAIN TRIUMPHANT*****"
1502 V$="&IEE   W!!N"
1504 W$="12111111211"
1506 GOSUB 5000
1510 PRINT
1520 PRINT "     I'D LIKE TO PLAY AGAIN SOMETIME, BUT RIGHT NOW, I"
1530 PRINT "HAVE TO REST!     BYE!!!!!'
1540 PRINT: GOTO 9999
2000 IF Y>=2 AND Z>=2 THEN 3000
2100 IF Y=4 OR Z=4 THEN S$="GAME": GOTO 4000
2200 IF Y=0 THEN Y$="LOVE - "
2300 IF Y=1 THEN Y$="15 - "
2400 IF Y=2 THEN Y$="30 - "
2500 IF Y=3 THEN Y$="40 - "
2600 IF Z=0 THEN Z$="LOVE"
2650 IF Z=1 THEN Z$="15"
2700 IF Z=2 THEN Z$="30"
2800 IF Z=3 THEN Z$="40"
2900 S$=Y$+Z$: GOTO 4000
3000 IF Y=Z THEN S$="DUCE": GOTO 4000
3010 IF Y=Z+1 THEN S$="ADD IN": GOTO 4000
3020 IF Y=Z-1 THEN S$="ADD OUT": GOTO 4000
3030 IF Y=Z+2 OR Z=Y+2 THEN S$="GAME"
4000 RETURN
5000 REM
5010 REM     SUBROUTINE WHICH "SAYS" THE CHARACTER STRING V$
5020 REM     PITCH CONTROL INFORMATION IS PASSED IN W$
5030 REM
5040 FOR II= 1 TO LEN(V$)
5050 C$=MID$(V$,II,1)
5060 U$=MID$(W$,II,1)
5070 WD=ASC(C$) AND ST
5080 IF U$="1" THEN WD=WD+SF
5090 X=USR(WD)
5100 NEXT II
5110 OUT 254,0: REM TURN OFF THE VOICE!
5120 RETURN
9999 END
```



TO BE CONTINUED

About Marsport....

I recently received the following paragraphs from Ned Sonntag, the talented creator of the Marsport cartoon strip which has appeared in Creative Computing *over the past year or so. For both new and old readers, the following will bring you more-or-less up to date. —DHA*



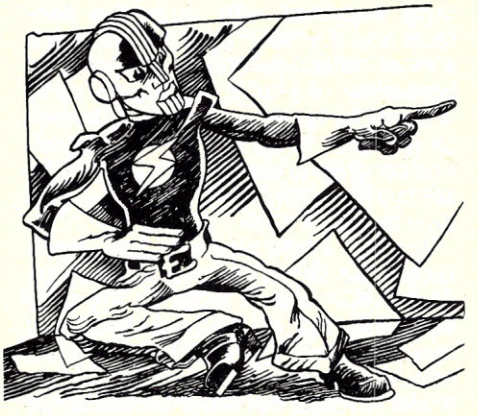
Let's see, this thing is very intuitive; but I started picking up futuristic impulses on my mental radio back around '73 ... in fact, these are the first two on this yellowed flyer I'm enclosing.

The duck in the hard hat is inspecting radioactive parts for defects. They are passing on a conveyer belt in front of a closed-circuit camera, and if he spots a defect he pulls the lever and the belt stops. The belt is somewhat like a Twist-O-Flex watchband. It develops a trapdoor and the cracked ring of plutonium drops back into the molten vat. The duck is a mutant born in 2001 on Earth who had to travel to the Martian colony to find work in late 2023. He has a wife (a chicken) and two bird-like children back on Earth who are forced to retreat to an underground leadlined city when a nuclear accident occurs in mid-February 2024. They are thus protected in fact because the duck works for Ohms Electric. This is a unique 21st Century corporation.

The duck works at Jaws Jarvas Rocket Repair, a body shop run by a brilliant, crusty native Martian. This is a division of Ohms Electric, a cybernetic corporation supplying electric power to the Martian colony. It is headed by a cyborg, "Spark" Ohms, a human brain in an android body. He is always devising ways to eliminate and replace the last human cells in his system. He is highly immune to radiation. More and more he can become one with his computers and experience the city as if it were his body : every electric terminus corresponding to a nerve. But there's much he doesn't know about the running of the city. Conspiracies of rich people run things in 2024. Technology leaps forward, new Andrew Carnegies, new Rockefellers . . . To give himself a human agent, Ohms revives a cryo-genically suspended Wrigley la Rock, with whom Ohms used to play in a band back in '73&'74. This gives him a crony also in his 70's but also like him untouched by age. This is the closest anyone in the 21st Century has come yet to achieving immortality.

Marsport is full of Art Deco skyscrapers, and everyone wears Mid-WWII-style clothes. An envelope of air surrounds the city, held in place by the same preservative that keeps Good Humor bars from melting on a summer day.

BEST, NED



A Creative Computing Equipment Profile. . .

OSI Challenger IIP

The Challenger IIP is up and running nearly as fast as it can be unpacked. All you need do is hook it up to any CRT monitor (or TV set using a RF modulator), plug in a tape recorder, and you're soon writing your first program.

Neil Shapiro

There are some new faces in computerland. They're the new, take 'em home and plug 'em in personal computers that seem on the verge of becoming true mass-market consumer items. With BASIC in ROM they speak the Queen's English (or at least Dartmouth's); they have their own keyboards, either self-contained or marketed right alongside; they're equipped with a CRT display or can use the family TV set; and if they are not supremely easy on the wallet (at around $600 to a shade under $1000), well, the price goes down smoother than anything their equal would have cost just two years ago.

The Challenger IIP from Ohio Scientific at $598 is the latest entry into this field, and it may be a real contender for the title. Thus far — what with the Commodore PET, Radio Shack's TRS-80, and the Heathkit systems — the Challenger IIP is just about the only "consumer computer" that can boast of a true computerish pedigree. Ohio Scientific is no newcomer to the computing field and had already established itself with other hobby and business systems before the introduction of the IIP. They manufacture, among many other items, the only hobbyist-level triple processor board and a 74-megabyte hard-disk drive.

How come the Challenger IIP? Their Fall '77 catalog spells it right out: "The Challenger IIP is a four-slot computer . . . designed for direct competition with the Commodore PET." There is an enormous potential market out there that, for some reason, non-computer-oriented electronic firms latched onto first. Now the computer companies are closing in and Ohio Scientific would love to lead the pack.

Neil Shapiro, 32-20 91st St., Apt. 607, Jackson Heights, NY 11369.

Has it worked? Has a "real" computer company managed to do everything right, surpassing all else in the field? That's a tough question to answer, but the Challenger IIP is certainly worth the careful attention of anyone who is now in the market for a hobby computer.

On The Outside

Ohio Scientific has packaged the Challenger IIP in an attractive cabinet which measures 15"x18"x4½". The general appearance is similar to the SOL-20 computer, but without the wood. It's certainly a computer no one would have to be ashamed to keep in the livingroom.

The self-contained keyboard is a dream to operate. It's a real typewriter-style keyboard with capacitive contacts and the "feel" to a touch-typist is almost indistinguishable from an office typewriter. If you've been annoyed in the past by hard-to-type-on keyboards (a real pain when that four-hundred-line program comes along), you will be pleasantly surprised the first time you type away on a IIP.

Nothing could be easier than interfacing the IIP with the human world. Three RCA phono jacks on the back panel take care of *video out, tape in* and *tape out.* The video is RS-232 and you may attach any CRT monitor. If you prefer to avoid that expense, use any of the low-cost RF modulators to hook it up to your TV set's antenna terminals.

Five minutes after you get it home, the Challenger IIP is up and running. Five minutes after that you should already have RUN your first program. What a far cry from the multitudinous hours of assembly and debugging that faced each and every hobbyist a few short years ago!

It Speaks Your Language

Ever since it was invented on Dartmouth's campus, the BASIC computer language has been a favorite in the hobby. It has its critics and detractors and no one would maintain that it does everything, or is even the best of the higher-level languages. Still, when most people talk to a microcomputer, they are talking BASIC. The Challenger IIP is equipped with a very well-designed 8K BASIC resident in on-board ROM.

The 8K BASIC was designed by the firm of Microsoft for Ohio Scientific and they have turned in a nice interpreter. This BASIC has just about everything you will probably need. If you are familiar with the Altair BASIC then you already know how to speak to the IIP. Microsoft designed both Altair's 8080 and 6800 BASICs. The major difference is speed. The Challenger IIP uses the 6502 microprocessor which, being faster than the 8080 or 6800, allows the IIP's BASIC to run about six times quicker than the Altair's.

There seem to be no bugs in this BASIC. It's just a nice, complete, dependable language that will get the computer to do what you tell it.

You can use two-letter variables, subscripted in arrays or matrices; there are the Boolean logic operators NOT, AND, OR; available are both trigonometric and logarithmic functions; there is a wide range of string-handling

functions; and the hobbyist is allowed direct access to computer memory via POKE and PEEK and to machine language subroutines via USR.

By no means, though, does the IIP "tie" you to using BASIC. Machine language is instantly available via the IIP's monitor. Any address can be programmed and it's all done hexidecimally right at the IIP's keyboard.

The Inside Story

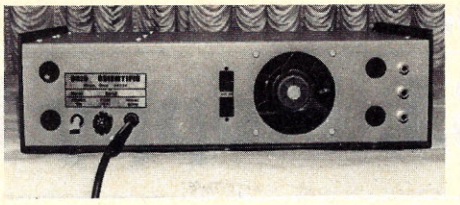
The Challenger IIP is based on Ohio Scientific's new Model 500 board which, in turn, is based on the 6502 microprocessor from MOS Technology. Many of OSI's larger systems also use this board (any of the Challenger computers with a Roman numeral II in their name).

The 6502 microprocessor is similar to the 6800, being the latest descendent of that chip. There are, however, differences that negate using 6800 software on a 6502. The main difference is in the stack pointer. Whereas the 6800's stack pointer is 16 bits wide, the 6502's is 8 bits and maximum stack length is 256 bytes. Memory is therefore often partitioned improperly in a 6800 program if looked at by a 6502. The index register is also different — the 6502 splits a 16-bit register into two separate 8-bit registers. You can get around these things but it would take considerable reprogramming.

However, software for the 6502 in machine or assembly language is *not* all that hard to come by. Many of the popular journals have published quite a bit. Also, as the Apple II and the Commodore PET computers are 6502-based themselves, it seems a safe bet that we will be seeing much, much more in the way of 6502 programs in the upcoming months. (Ohio Scientific publishes a "Small Systems Journal" which is one of the livelier user group newsletters in the industry, and it carries an abundance of 6502 programs and programming techniques. The Journal is free to new Challenger owners and $6 for six issues to other subscribers).

The Challenger IIP's 500 board comes complete with 4K of 2102-type RAM memory chips, along with four 2616-type ROMs which contain the BASIC language. It is a versatile board: the 2616 ROMs may be replaced with 2704s, 2708s, or 2716s if the user wishes to add his own custom software. Though it arrives configured for a video RS-232 operation, it can later be changed to 20-ma loop. Up to three 1702 PROMs may be added to the board and there is a 256K Memory Management option (this allows the computer to address up to 256K of memory).

The 500 board occupies the first slot of the Challenger IIP's four-slot motherboard (which OSI calls a backplane). The bus is a 48-line arrangement and the IIP will accept much of OSI's accessory boards. (A slight problem: the IIP's power supply is shy of +12 volts — but more on that in a moment).



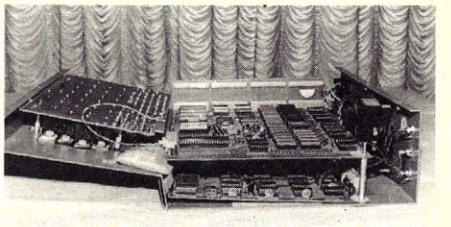
Interfacing the IIP is quite simple. The three RCA phono jacks on the back are for video out, tape in and tape out. The computer is also supplied with a fan to keep the innards cool.

The second slot of the backplane — leaving two slots open for expansion — is taken up by the 540 video board. The video board can display *either* 32 lines of 32 characters, or 32 lines of 64 characters; a simple POKE command at the keyboard instantly changes back and forth from each format.

While the 540 display is, of course, normally used as a conventional CRT display board; the entire display memory is accessible as normal memory to the computer. When you program, you may directly address any portion of the display. On-screen animation, constantly up-dated displays and more are all possible through use of this feature.

The cassette interface is one of the easiest to use. It is simply a matter of typing SAVE (then LIST) or LOAD. Running at 300 baud, the computer LISTs out, on the CRT or RF-modulated TV, each line of the program as it is being recorded. It also visually LISTs a recorded program that is being played back into computer memory.

Cassette operation, however, is completely under manual control. For instance, the computer will not turn off the recorder when a program is through being read in; it is up to the human operator to press "stop" on the recorder. It is a minor inconvenience but, according to a technician at Ohio Scientific, this mode was chosen so that "a person would be able to use any moderately priced recorder, including those without Remote functions." Still, it would have been nice to have had automatic control.



The 500 board takes up one slot of the motherboard and beneath it, in a second slot, is the 540 Video board. This leaves two slots open for further expansion. Due to the 48-line bus, the IIP will only accept boards manufactured by OSI, but a number of options are in production for the IIP.

Baud rates are jumper-selectable up to 600 baud on the parallel-serial/serial-parallel part of the interface, located on the 500 board. The analog-digital/digital-analog portion is located on the 540 board.

There we have two slots of the four-slot backplane filled. What about the other two? After all, nature and the computer hobbyist abhor a vacuum. What does Ohio Scientific have in mind...?

Looking Ahead

With a *Star-Wars* name like IIP (wonder if R2D2 and C3PO got their start this way?), it should come as no shock that a future is planned for this Challenger. According to Ohio Scientific, the IIP has enjoyed so much "overwhelming acceptance" that many accessories are now in the planning stages, and a few have already begun production — likely to be available by the time you read this.

First off: memory. As now manufactured, the IIP can accept Ohio Scientific's 4K board but not the 16K. The reason? As mentioned previously, the IIP's power supply just does not supply the +12 volts required by the larger board.

OSI will soon release a new 16K board, designed specifically for the IIP, which will have an on-board inverter to obtain the +12V. It's too bad that the IIP's power supply wasn't designed at the start to overcome the problem. However, once the new board comes on the market, the IIP's in-case expansion will be 36K.

There will also be an expansion chassis for those who feel they could use even more memory or goodies. It will essentially be one of the standard eight-slot Challenger cases. The user will only have to move his Challenger IIP 500 and 540 boards to the new chassis and then run a ribbon cable to connect the IIP's captive keyboard to the chassis.

Then there are plans for a "low-cost" and full-size, eight-inch floppy-disk drive. The disk will be presented in a case to match the IIP and will contain a built-in power supply. Though plans could change, right now that disk drive is planned to be compatible with the IIP's ROM BASIC. If so, the addition of the disk should be painlessly easy.

If you are in the market for a hobby computer, and you'd like the convenience of a captive keyboard with BASIC in ROM all at a low cost, you should consider the Challenger IIP. Its features and performance make it interesting indeed. ■

A Creative Computing Equipment Profile. . .

M.S.I. Floppy Disk

Steve North

It is generally acknowledged that a floppy disk is required for sophisticated data handling and storage with a microcomputer. Midwest Scientific Instruments' FD-8 floppy disk memory unit contains a G.S.I. Model 105 or 110 disk drive, and a controller that may be interfaced with any system having two bidirectional data ports. However, the FD-8 is primarily for use with M6800 - based systems, such as the Southwest Technical Products 6800 or M.S.I.'s own system, since most of the options and software supplied for the FD-8 are compatible with 6800-based systems.

The floppy-disk controller in the FD-8 is not intelligent since it relies on the host computer for major functions of its operation. This isn't really a disadvantage, but merely means that designers of the FD-8 made a tradeoff — more software and memory requirements in return for less hardware. CPU time isn't usually at a premium in personal computer systems, and anyone willing to buy a floppy disk should also be willing to buy another memory board. The controller has a sector buffer so that I/O operations may be done independently of the speed of the CPU. Hard-sectored diskettes (not IBM standard) must be used. The FD-8 is configured for 256 bytes per sector, 16 sectors and 77 tracks per diskette, for a total of 315,392 bytes of storage per diskette. Under normal conditions a handful of these tracks are reserved for use by the system, but the rest are available to the user. The controller may be jumpered for 32 sectors per diskette with 128 bytes per sector, but M.S.I.'s software does not support this format.

As we mentioned before, the FD-8 is interfaced with two bidirectional parallel data ports (provided by a single PIA chip). M.S.I. sells an interface card designed especially for use with the FD-8, but other interface boards could be used.

M.S.I. supplies a Floppy Disk Operating System for use with the FD-8. To be used, the FDOS must be read from the system diskette into memory at 2400 hex. There are three ways to do this. First, M.S.I. supplies a disk bootstrap program on a cassette. This program is loaded and executed at 2400 hex, and then it loads the FDOS from the diskette and executes it. The same cassette also contains a program called MINIDOS which is, as its name suggests, a mini-floppy disk operating system. MINIDOS can be used to read or write sectors on the diskette to or from memory in the computer. So, in its crude way, MINIDOS could be used to read the FDOS from the diskette into memory at 2400 hex. MINIDOS must be loaded starting at location 7700 hex, where most people don't have any memory. The third method (which we chose) is to get a PROM board with a disk bootstrap program on it. M.S.I. sells just such a PROM board, and supplies the disk bootstrap program in two 1702A EPROMs. Actually, we're a bit surprised that they used 1702As, because 1702s are too slow for the 6800, and 1702As are just barely fast enough. But it does work, and it's much faster and more convenient than bothering with cassettes every time you want to use your disk. Using a disk bootstrap PROM board, one merely tells the MIKBUG monitor to execute the bootstrap program located at C000 hex, and you're off!

The M.S.I. FDOS is easy to use. Commands are entered as keywords, such as LOAD, SAVE, RUN, COPY, and CATALOG. These commands permit you to do such things as load and save files, print a directory, attempt to recover damaged portions of a diskette, initialize a new diskette, etc. File names may contain up to six characters. We did note a rather nasty problem with the FDOS: it does not check for duplicate file names when you ask to create a new file. In other words, you are permitted to create 27 files with the same name. Subsequently, you can only access the file that was created first, because that's the file the FDOS will see first when it looks in the directory. At any rate, the FDOS supports the following file types:

System Files: System files contain programs that are considered part of the operating-system software. This includes BASIC, assemblers, a text editor, and utilities (such as one to PACK the diskette to recover space taken up by deleted files, and another that lists the directory including passwords and system files.) To run a system file, merely type its name, such as BASIC, and the program is loaded and run. It is also possible to create your own system files, by making the first character of a file name a dollar sign. The file is then considered to be a system file. This is a nice feature, but unfortunately it is not possible to delete a system file, so it must be used carefully.

Text Files: These are files created by the co-resident text editor/assembler. They may be assembly-language programs or just general text.

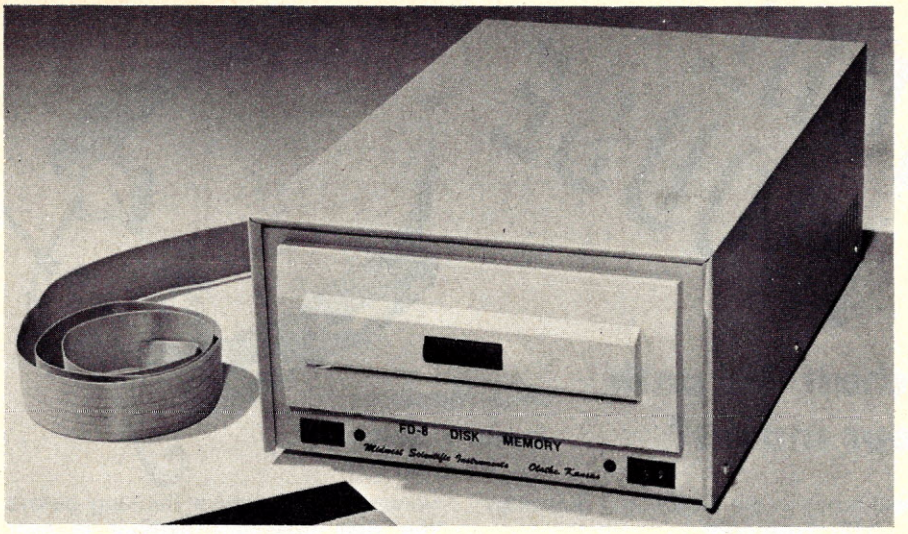
Object Files: Output produced by the assembler.

Machine Code Files: These are files you create by directly saving a portion of memory on the disk. An example might be an old version of BASIC, which was previously loaded from audio cassette before you had a floppy disk.

BASIC Programs: BASIC programs may be saved on the diskette, both in a "packed" and in a pure ASCII format. In the packed format, the keywords are condensed into BASIC's internal format. In pure ASCII format, the program is saved on disk exactly as it appears when it is listed. The reason for the difference is that a packed program takes up less space on the disk and may be loaded more quickly, but an ASCII program can be appended to a program already in memory. Additionally, programs saved in pure ASCII would be compatible with a text editor. At present, the text editor in the co-resident assembler/editor isn't much different than a BASIC editor. However, M.S.I. plans to come out with a much more sophisticated text editor in the future. When they do, it would be very handy to be able to edit BASIC programs with the fancy text editor.

BASIC

Since most personal computer users want to use a high-level language with their systems, we'll take a closer look at M.S.I.'s Disk BASIC Interpreter. M.S.I.'s BASIC is based on the 8K BASIC written for the 6800 by Robert Uiterwyk. However, the POS and SGN functions have been removed. In return, there are many more useful features, including TRACE, ? as an abbreviation for PRINT, and most importantly, statements, commands,



and functions for handling programs and data files on disk. M.S.I. Disk BASIC takes up nearly 16K of memory, so 24K or even 32K are required to run reasonably large programs.

Saving and loading of programs is accomplished in a straightforward manner with the LOAD, SAVE, REPLACE, and CHANGE commands. The CHANGE command is used to save a program in ASCII format, so the command is not really very suggestive of its function. Well, maybe they were running out of command keywords. Data-file manipulation is somewhat more complex. A CREATE statement (or command) is used to reserve space on a disk for a data file. Then, an OPEN statement must be executed to access the file in a program. The OPEN statement assigns a "channel number" to the file, so that the file is subsequently referred to by number, not name. When the file is opened, it may be accessed for INPUT, OUTPUT, or UPDATE. I/O is done in fixed-length records. A FIELD statement is used to define the record format for a file. For instance FIELD #10, A$=40, C=5, X$=20 means that when a record is read from file #10 (which was OPENed before), the first 40 characters of the record will be put in variable A$, the next five will be put in variable C (a number), and then the next 20 characters go into variable X$. So, the total record length is 40+20+5, or 65 bytes. GET and PUT statements are used to do input and output with a data file. With the example of a FIELD statement used above, GET #10 would read a record from file #10 and place the information in the appropriate variables, while a PUT would have the opposite effect. The OPEN for UPDATE feature is interesting. When a file is opened for update, one may change records of the file in place, using a REWRITE statement (similar to GET and PUT). Note that at any time no more than three files may be open, and only one for OUTPUT.

Random access within data files is provided by a SET statement, which permits you to set the file pointer to any record within the file. For example, SET #20=N would set the file pointer to the Nth record in the file. LOC# is used to determine the location of the file pointer, and EOF# is used to detect end-of-file conditions. Finally, a CLOSE statement is used when processing of a file is complete.

Overall, the facilities in M.S.I. Disk BASIC for processing data files are complete and most people would find them adequate for their applications. There does seem to be some overlap in the functions of the OPEN for OUTPUT and CREATE statements. Both OPEN for OUTPUT and CREATE can cause a data file to be created on the disk, and since the FDOS allows duplicate file names, you can end up with a rather unpleasant problem. M.S.I. Disk BASIC also has program CHAINing (which we feel *all* disk BASICS should have) and a means for one program to call another from disk, and for the subprogram to return to the next statement in the main program when completed. A nice touch, we thought.

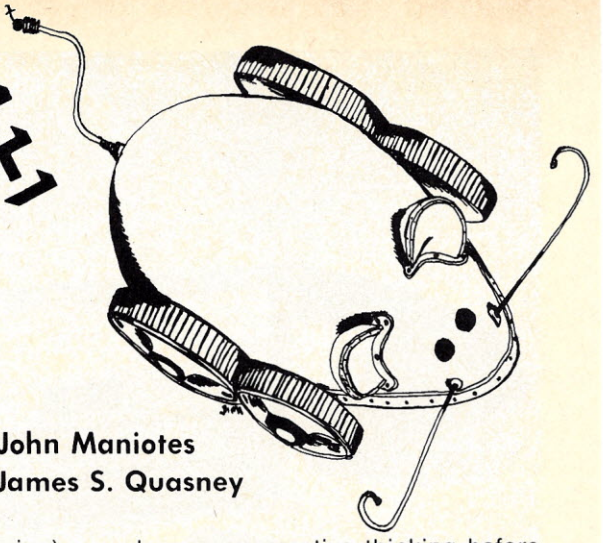
M.S.I. also plans to introduce a BASIC compiler (not an interpreter) which will data files, PRINT USING (for formatting of output), and an ON ERROR option which permits trapping of errors within a program, rather than have your nice application program go BLA! in front of the dumb user. The compiler is a two-pass compiler, which produces an assembly language version of your program. Then you assemble the program using the disk assembler. According to M.S.I. the compiler is much faster than the interpreter. For instance, a card shuffling program which ran for 90 seconds under the interpreter ran in three or four seconds using the compiler. Indications are that the compiler used with a 1.6-MHz 6800 CPU, is faster than most of the other BASICs available for microcomputers. This should be an interesting product, especially for business applications.

Summary

The FD-8 manual contains a step-by-step construction list, schematics, drawings, calibration instructions, an explanation of how the FD-8 works, and documentation for the software including source listings of the disk drivers; and diagnostic routines on a cassette tape as well. We have found the FD-8 to be a reliable product. At one time we did have some trouble, when we made a factory recommended modification to prevent accidental write operations (a very bad thing if it happens—we had no problems, but we decided to make the modification anyway). We botched the job and managed to get a tiny sphere of solder between two pins on the connector which runs from the disk to the computer. The result was that the disk would operate normally for about 30 seconds when it was turned on, and was then unable to step the head back towards track zero after it had been moved out. We called M.S.I., and they immediately suggested that the connector on the FD-8 might have been shifted to the side, causing two pins to be shorted together, which made a driver get very warm and also not work very well. We went back and checked the connector on the FD-8, which was OK, but armed with this information we decided that there was a short and found it readily.

The big question for those considering purchase of a floppy disk for their 6800-based system will be whether they should opt for the FD-8, or some other product (such as Southwest's dual minifloppy, or Smoke Signal's triple minifloppy). There's no clearcut answer, and we haven't had any experience with the other units, so we won't make a recommendation. The minifloppy units get only 90 or so Kbytes on a diskette, which isn't that much, considering that a few big files and some system software will easily chew that up. It is true that both SWTPC and Smoke Signals get you up and running with a floppy for less money than M.S.I., and that it is sometimes *very* handy to have more than one drive (for copying files, processing one file against another, etc.). Of course, M.S.I.'s controller can handle up to four drives by daisy-chaining, and two full-sized floppy drives will outperform two minifloppies, with only a small difference in cost, like only 100% or so. Also, we hear that the SWTPC and M.S.I. disk operating systems and BASICs are similar. Ultimately, price/performance will be the deciding factor for both hobbyists and application-oriented users. ■

THE MECHANICAL MOUSE



Can you flowchart his path through the four mazes?

John Maniotes
James S. Quasney

Flowcharting "mechanical things" has been around for quite a long time in beginning programming courses. A popular flowchart problem, which the senior author was exposed to in the late 1950's and which has since undergone many revisions, is *The Mechanical Mouse* problem. This is a fun-type flowchart problem that should delight the novice, intermediate, and professional programmer.

The Problem

Draw *one* flowchart that will cause the Mechanical Mouse to go through any of the four mazes shown in the figure. At the beginning, an operator will place the mouse on the entry side of the maze, in front of the entry point, facing "up" towards the maze.

The instruction "Move to next cell" will put the mouse inside the maze. After that, the job is to move from cell to cell until the mouse emerges on the exit side.

If the mouse is instructed to "Move to next cell" when there is a wall in front of it, it will hit the wall and blow up. Obviously, the mouse must be instructed to test if it is "Facing a wall?" before any "Move."

The Mechanical Mouse's instruction set consists of the following:

A. Physical Movement
   (1) Move to next cell (the mouse will move in the direction it is facing)
   (2) Turn right
   (3) Turn left
   (4) Turn around (all turns are made in place, without moving to another cell)
   (5) Halt.

B. Logic
   (1) Facing a wall? (Through this test, the mouse determines whether there is a wall *immediately* in front of it; that is, on the border of the cell it is occupying and in the direction it is facing.)
   (2) Outside the maze?

If the mouse is outside the maze, the mouse can also make the following decisions:
   (3) On the entry side?
   (4) On the exit side?
   (5) Branch (unconditional to any part of the program).

Types of Solutions

There is a variety of ways of attacking this problem and a variety of solutions.

Beginners seem to use two methods of *attack* to gain a solution. The first involves the "sledge-hammer" approach, where a flowchart is written to work for one maze and then additional logic is added in a piecemeal fashion to handle the remaining three mazes. Naturally a lot of trial and error is involved, and the flowchart solution is spread over several pages, making it difficult for one to comprehend the solution readily.

The second method of attack involves some creative thinking before the first flowchart symbol is ever drawn. The key centers around the definition of a *cell*. In this problem a cell is a "four-sided" figure with one or more sides missing. It is this symmetry that one wants the mouse to take advantage of so it can turn right or left or around accordingly.

The types of flowchart *solutions* generally fall into the "short" or "long" flowchart category with some solutions in between these two extremes. The short flowchart solutions have a few symbols (six to seven symbols, excluding Start and Halt) but subject the mouse to a lot of false and inefficient turns in each cell.

The long flowchart solutions have a lot of symbols (15 to 20) and subject the mouse to few false and inefficient turns in each cell. Other flowchart solutions are in between these two extremes and represent a compromise.

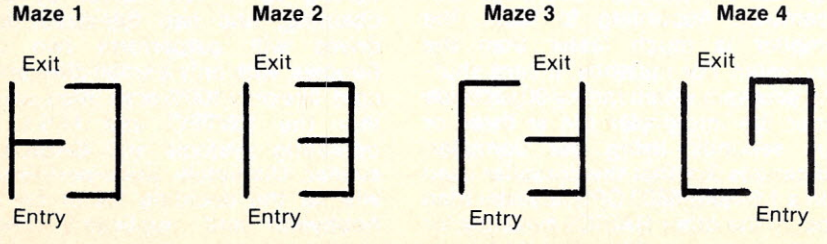
The short flowchart solutions have the advantage of using less "storage" than the long ones. However, the long flowchart solutions take less "execution time" for the mouse to carry out its objective. Hence, one has to weigh the amount of "storage" and "execution time" used to determine the "best" solution.

Note that one flowchart solution must work on all four mazes. The hardest maze for the beginner is usually maze 4. So don't be surprised if your flowchart works for the first three mazes but fails on the fourth maze.

As an extra-credit problem, enlarge each four-cell maze given to either a 9 or 16-cell maze and see if your existing solution still works for the new mazes as well as those shown in the figure.

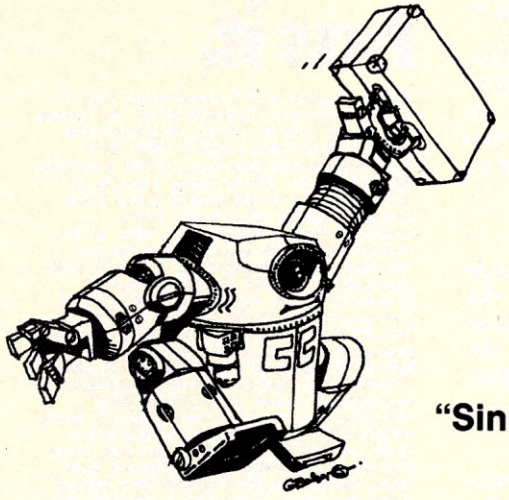
For those who desire a solution to The Mechanical Mouse problem, please send the senior author a self-addressed stamped envelope (SASE) and enough postage for its return. For those who have other versions of this problem, we would be interested in corresponding with you. Either way, we hope you have fun with The Mechanical Mouse problem! ■

John Maniotes and James S. Quasney, Information Systems and Computer Programming, Purdue University, Calumet Campus, Hammond, IN 46323.



Four mazes where each maze has four cells.

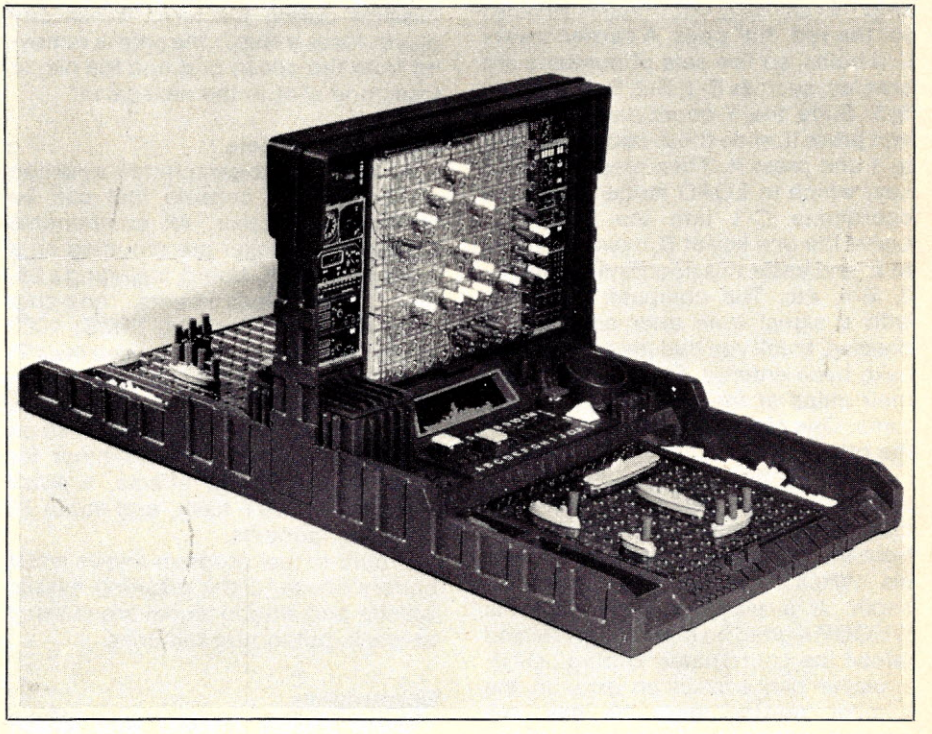
Profile of a Smart Electronic Game . . .



Electronic Battleship

"Sink your opponent's fleet using pushbutton warfare."

Stephen B. Gray



The ad shows a couple playing the game; he looks shocked; she smiles. The large type reads "Great Sea Battle on East 78th Street!" The smaller print says, "Only skill, daring and luck has kept surviving ships afloat, the outcome in doubt. Now—in one inspired move—it can all end. The excitement mounts. The whistle of shells rips the air, explosions flash and rumble. The last enemy ship sinks in a blaze of battle sounds and sights. New Electronic Battleship is so real it's unreal. You plot strategy, defend your fleet, and destroy your opponent's fleet by computer logic. And victory is sweet. New Electronic Battleship, the exciting computer strategy game!"

The Box

This is the largest of the electronic games we've profiled so far; the box is almost two feet long, a foot wide, and six inches high. On opening the box, you find a lot of cardboard has been used to protect Milton Bradley's electronic version of their older all-plastic game.

Electronic Battleship is available at game and toy departments, and at toy stores, for $30 to $50.

Object of the Game

According to the manual, "Electronic Battleship is a computerized naval battle game for two players. Be the first to sink your opponent's fleet using pushbutton warfare. You program the location of each ship into the computer and fire missiles at your opponent's ships while the computer records the battle with realistic sounds of probing sonar, flying missiles and shattering explosions."

Box Contents

The long base unit consists of two computer control consoles, one for each player. Each of the two 10-by-10 ocean grids is accompanied by a pair of sliding coordinate keys that enable a player, during setup time, to enter into the computer the exact locations of his five ships. A single LOAD/GO switch is placed in LOAD position "when you are entering the coordinates of your ships into the computer," and in GO position "when both players are ready to begin the battle." There are only three other switches: a FIRE key for each player, and an ON/OFF switch. At the rear of each console is a ship silhouetted against a red screen; this is for... but let's wait until later for that.

The other main part is the vertical Target Grid superstructure, which slides onto the base, and which divides the playing into two "secret zones" so that a player can't see how his opponent's ships are deployed. The superstructure also has a target grid on each side, identical to each player's ocean grid, but used for keeping track of hits and misses on the opponent's fleet. The third item in the box is a plastic bag containing two sets of 84 white pegs (they indicate misses), two sets of 42 red pegs (to indicate hits),

and two each of these ships: battleship, destroyer, carrier, submarine, and PT boat.

Setting up the Fleet

Once the computer has been informed of the locations of each player's fleet, the game goes fairly fast. A little time is required at first to give the computer the coordinates of each point occupied by each of the five ships on each side.

Move the ON/OFF switch to ON; "you will hear the BEEP...PING sound of the sonar" coming from a two-inch speaker inside the console. Each player moves his X-coordinate key to the CM position and presses it to clear the computer's memory of his previous ships' positions.

Place a ship on the ocean grid by pressing the anchoring pegs under it into the grid holes. Move the LOAD/GO switch to LOAD, and enter the ship's coordinates into the computer. Each ship has as many coordinates as holes for the red "hit" pegs. A carrier covers five holes, so five sets of numbers are entered, such as B-1, B-2, B-3, B-4, and B-5. Slide the Y-coordinate key to B and press it; slide the X-coordinate key to 1 and press it. Then press the FIRE key, which in LOAD mode enters the coordinate C-1 into the computer. Leave the one key at B, move the other to 2, and enter this coordinate, then B-3, B-4, etc. The computer responds with a signal tone after each key is pressed, to tell you that the coordinates have been entered. Players may enter their ships at the same time, or take turns. One player's signal tone is high; the other is low.

When all five ships and 17 coordinates are entered correctly, each player will hear his own WHOOP signal from the speaker, and each must hear his WHOOP before the battle can begin. If a player doesn't hear his WHOOP signal, he must press CM and reload the coordinates. During LOAD, a player can correct an error in the number and/or letter he has just entered, without having to erase all the previous entries, by sliding his Y-coordinate key to the CLE (Clear Last Entry) position and pressing it.

To Fire a Missile

The first shots are guesses. You choose a coordinate on the upright target grid, put a white peg at that point, set your coordinate keys to that grid point, press the keys to enter the missile coordinate into the computer, and press FIRE. The computer will respond with the WHISTLE sound of a missile in flight.

If you score a HIT, the computer will cause a flash of light on the red screen and the sound of an explosion. So you take out the white peg and put in a red one, to record your hit. You also tell your opponent the coordinate of the section of his ship you've hit. He must tell you which ship was hit (carrier, PT boat, etc.) and must place a red peg in the corresponding hole in that ship.

Now you have to figure out where the rest of that ship is. If you had a MISS, there is no flash of light and no explosion, and you just leave the white peg at that coordinate on your target so you won't aim there again.

Winning

When any ship is HIT enough so that all its holes are filled with red pegs, the ship is sunk. The first player to sink all five of the opponent's ships is the winner and will hear the Victory signal: WHOOP...WHOOP...WHOOP.

Salvo Game

"The Salvo variation is for experienced players who are familiar with the basic game." Each player fires five missiles during his turn. "When any player loses a ship...the ship is removed from the ocean grid and the player loses one shot in the next salvo."

Inside the Console

All controls are centered on a portion of the control console that can be removed for repair. All components except the speaker are mounted on a 5½-by-6-inch PC board, including a 28-pin Texas Instruments one-chip microprocessor of the TMS 1000 series. This family of ICs is used in all TI calculators, as well as in another Milton Bradley game, COMP IV, reviewed in the Nov-Dec 1977 issue (p 36). Also on the PC board is a TI 555 timer for clocking the sound effects, a small lamp for the HIT flash, and about 70 other components.

A built-in test program allows a fast battery check: if the resulting missile whistle and whoop signal are missing or weak, put in new batteries.

Observations

Game-players familiar with Milton Bradley's all-plastic Battleship game, which has been a favorite for years, will be playing the electronic version in no time at all, because all the rules, strategy and game play are identical with the older version. Although a few minutes are required to input the 17 ship coordinates to the computer, that time is quickly forgotten amidst the continuous sonar bleeps, the whistle of incoming missiles, the HIT explosions, and the whoop of victory when the entire fleet is sunk. If that fleet is your opponent's, then victory can indeed be sweet. But if it's your fleet that went to the bottom, try to tell yourself it's only a game, stop pounding your head against the wall, and reset for a return match. Players, man your consoles! ■

Seven points to consider before you buy your small computer.

In this magazine, alone, there are probably a dozen ads for small computers. New companies are breaking ground like spring flowers.

How, then, do you determine which computer offers the features you need most... at the price you can afford?

We'd like to propose seven basic questions to help you make an intelligent decision.

1. How complete is the computer system?

Many buyers of small computers are in for a rude awakening when they have to spend additional money for interfaces.

The Sol-20 Terminal Computer was the first *complete* small computer system. Everything you need to make it work is included in the basic package.

2. Is powerful system software available?

It won't do if your system is "tongue-tied."

Processor Technology Corporation has devoted more effort to the development of software than any other small computer maker. Our latest offering is the first fully implemented disk operating system for a small computer: PTDOS. It contains over 40 major commands, several languages and numerous utilities. Our high level languages include Extended BASIC, Assembler, FORTRAN,* FOCAL and PILOT.*

3. Is the system easy to expand?

More and more computer owners are expanding their small computers to handle business and other specialized requirements.

The largest Sol system can handle 64K bytes of RAM memory and operate with a three megabyte on-line disk memory. Sol systems use the S-100 Bus. So you can use a wide variety of hardware.

4. Is the computer well-engineered?

Our Sol systems are the most conservatively rated and ruggedly built in the industry, period. In addition we designed them with you, the user, in mind; Sols are easy to build and a joy to operate.

5. Does it have proven reliability?

What is the track record? There are over 5,000 Sol systems in the field. Our track record for reliable performance is unparalleled in the small computer field.

6. Does it have good factory support?

A computer is a complex piece of hardware. So you want to be sure it is backed up with complete manuals, drawings and a factory support team that cares.

Processor Technology offers the most extensive documentation of any small computer manufacturer. And we maintain a patient, competent telephone staff to answer your questions.

7. Are maintenance and service people accessible?

Where are they located?

Processor Technology has maintenance and service people in over 50 cities around the U.S.

As you continue turning the pages, see how we stack up to the other computers in this magazine. If we've succeeded in whetting your appetite, see your Sol dealer or write for information on the complete family of Sol computers.

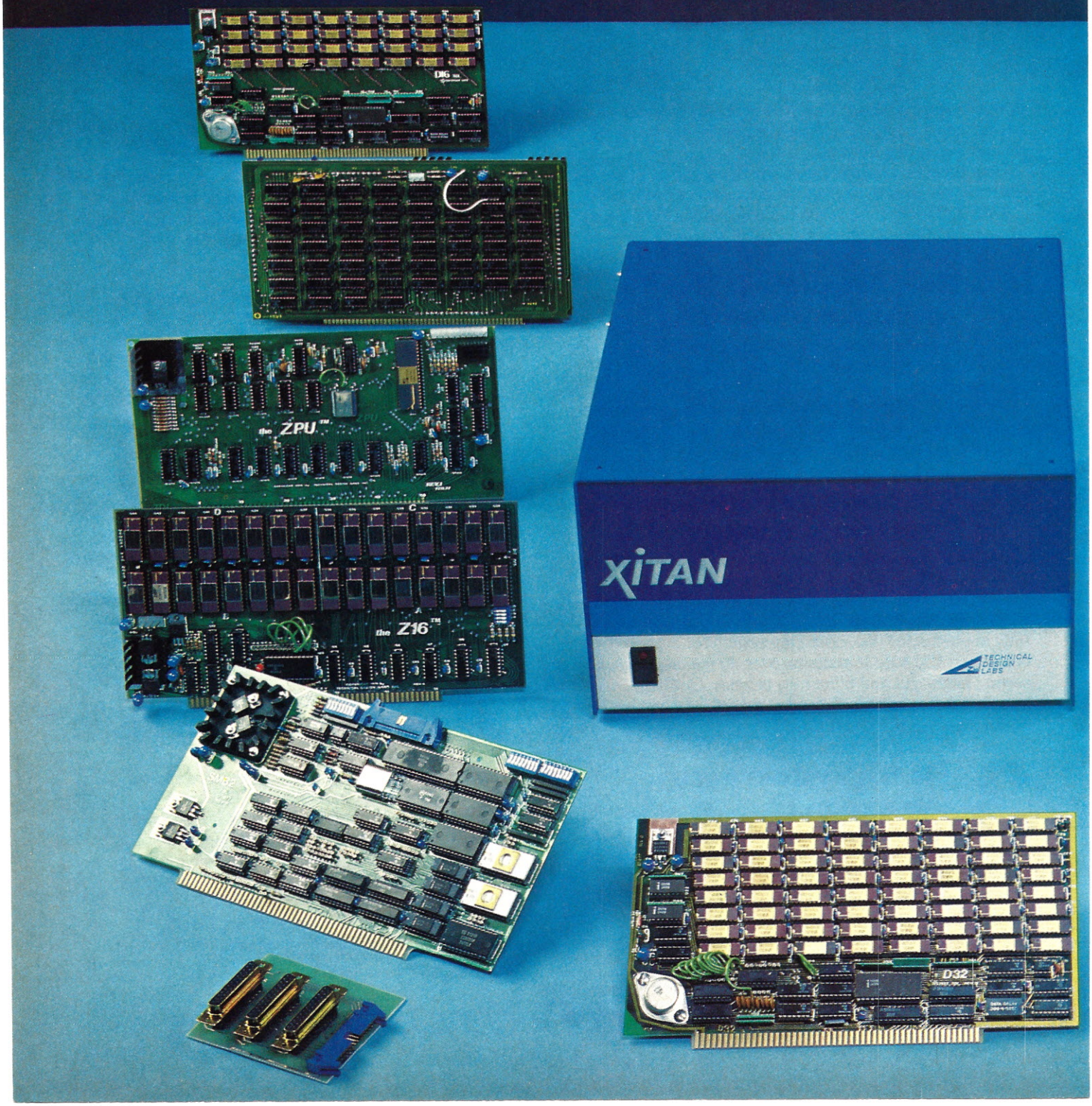
Processor Technology Corporation, Box C, 7100 Johnson Industrial Drive, Pleasanton, CA 94566. (415) 829-2600.

*Available soon.

ProcessorTechnology

CIRCLE 107 ON READER SERVICE CARD

The Ultimate in Microcomputer Hardware & Software



Systems

TECHNICAL DESIGN LABS INC., RESEARCH PARK, BLDG. H
1101 STATE ROAD, PRINCETON, NEW JERSEY 08540
(609) 921-0321

CIRCLE 111 ON READER SERVICE CARD

Umtech VideoBrain

David H. Ahl

VideoBrain comes with keyboard console, two plug-in joysticks, AC adapter, TV-antenna switchbox, and several program cartridges.

The VideoBrain is essentially a cross between top-of-the-line video games and full computer systems such as PET or TRS-80. VideoBrain has an F-8 microprocessor in it, 1K bytes of RAM memory and 4K bytes of ROM memory. It hooks up to a TV set, preferably color, like any of the video games through a TV/game switch box. It is powered by a low-voltage power adapter which is included. The unit itself measures 14 inches by 11-1/2 inches by 5 inches high. It comes complete with two plug-in joysticks. The unit has a 36-key keyboard which is a somewhat abbreviated typewriter keyboard. The number keys are in a numeric keypad arrangement at the left side of the board, which is somewhat unexpected but okay once you get used to it. The only thing I didn't get used to in playing with the unit over a period of time was that the key used as the equivalent to the RETURN in a computer — called on this unit the NEXT key — is at the bottom center of the keyboard and is the same size as every other key. Also, the second (upper-case) symbol on the keys correspond to no known pattern; that is, the T has the times sign on it, the Y the division sign, U has an exclamation point, O the number sign and so on. On the other hand, if you haven't been exposed to alternative layout over a long period of time you shouldn't have too much trouble in getting used to the keyboard.

Seven Lines of Print

The built-in memory of the unit provides for alphanumeric characters in the rather coarse arrangement of seven lines of print with 16 characters on each line. Like the programmable video games, this unit uses plug-in ROM memory cartridges. A ROM equipped with enough memory will allow for more tightly packed print characters up to a maximum of 16 lines containing 24 characters each. In contrast to the somewhat coarse print characters, the graphics are very good. The grid is 140 horizontal lines by either 200 or 400 vertical lines. In addition to the VideoBrain console itself, there is a matching expander unit that has a 3870 single-chip microprocessor in it, a 2K ROM memory and which allows the control of two standard cassette-tape recorders to store and retrieve data. It also has a third cable, to connect with standard RS-232 printers or communications devices and it has a cable with a plug for current-loop communications with a standard Teletype or acoustic coupler. Obviously the VideoBrain, when equipped with the expander box, is the equivalent of a small computer.

Program Cartridges

As of this writing, (March 1978) there are seven cartridges of programs available, with four scheduled for release within the next month or two. According to Ted Haynes, Manager of Product Marketing, many other programs are in the programming and/or planning stage. VideoBrain clearly is coming up with a somewhat different programming philosophy than the manufacturers of the video games. From the initial offerings, it is obvious that they are concentrating on somewhat more serious applications in

You can play checkers at three different levels with this cartridge.

the areas of household management and education. That is not to say that there are not some very good games being offered, but there are proportionately more education and household management programs than are offered by other manufacturers. Some of the initial offerings are somewhat simplistic in their approach but that should, of course, improve over time. Let's look at some of the program library.

Finance One

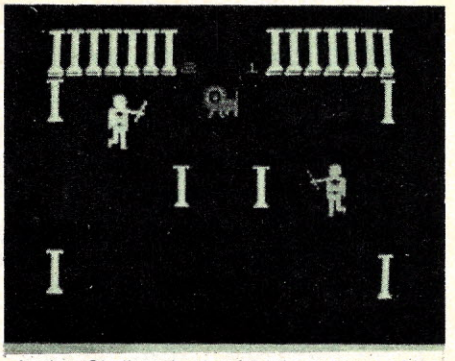
Finance I lets you complete analyses of loans, mortgages, savings accounts, and other financial alternatives. The program prompts you for the necessary inputs and then computes the value requested: net present value, internal rate of return, accumulated interest or principal for any period, interest rate or term of loan. The program also graphs the results of the last eight evaluations. In addition, you can write formulas for evaluation with up to 20 user-defined variables.

The VideoBrain is essentially a cross between top-of-the-line video games and full computer systems such as PET or TRS-80.

Music Teacher 1

This cartridge teaches the user to read, write, and play music in a four-octave range. When you play a note on the VideoBrain keyboard, the note is played through the TV, and the note is named and displayed in five-line musical notation on the screen. Once you've played a tune, the VideoBrain will play it back just as you keyed it in. The program also includes two built-in songs.

The only problem I had with this program is that songs are played back in the same tempo as when you type them in. Clearly, if you are used to the keys of the piano, typing a song in on a typewriter keyboard is somewhat foreign, and it is rather difficult to type it in real-time. It seems to me it would have been desirable to have a variation which allowed picking out notes and then having the computer put them together and play them back in tighter sequences than you might have typed in.



Ancient Gladiator is one of the three games, with a total of 384 variations, in the Gladiator cartridge.

Wordwise 1

This program trains from one to four people in word-building skills at 3 different skill levels. The computer gives each person a random assortment of ten letters and challenges the person to build words with them. The program includes a challenge round (to correct score for misspelled words) and a song to salute the best word-builder.

Wordwise 2

Wordwise 2 teaches accurate touch-typing through three exercises. VideoBrain signals your mistakes and records your progress with a words-per-minute score after every exercise.

On the same cartridge is Cypher, an electronic word game for two players. One player decides on a quotation or phrase to be scrambled by VideoBrain. The other player tries to unscramble it as fast as he or she can. Good scrabble players should have no trouble with this one. Although let me tell you when you are trying to unscramble a phrase of seven or eight words, it is no picnic.

Gladiator

Gladiator includes three games with an incredible 384 variations. In Ancient Gladiator, you aim your bow and arrow to fend off a hungry lion as well as your opponent. In Modern Gladiator, you can run or pass to two receivers to score a touchdown in "Scrimmage." Future Gladiator pits two laser-armed space ships against each other in an intergalactic battlefield. The strategic nature of each game changes as you add bouncing arrows, joystick-guided lasers, bullet passes or other exciting variations. If no opponent is available you can let VideoBrain play one gladiator while you run for cover with the other.

Blackjack

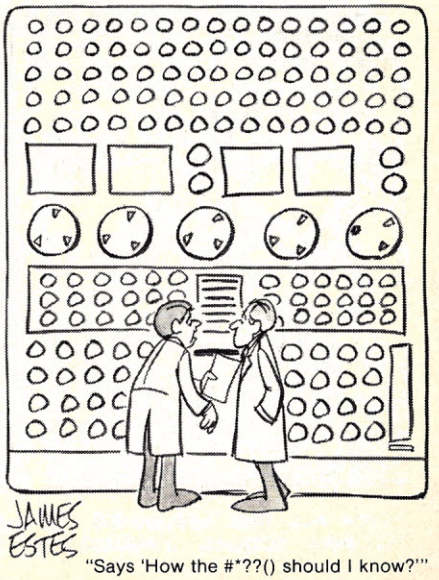
One or two players can try to beat the dealer in this Nevada-style game. Start with $500 and bet up to $250 with a move of the joystick. Blackjack pays you 1½ times your bet and you can double your bet if you're dealt a 10 or 11, so you can count cards to gain an advantage. VideoBrain plays a musical tone to tell you if you won or lost.

Other program cartridges scheduled for early release include: cash management, real-estate analysis, stock valuation, math tutor, pinball, checkers, and tennis.

With the Cash Management program, you keep permanent records of all household income and spending.

How Much and Where

The price of the basic VideoBrain $499 and the expander unit is $199. ROM program cartridges cost $20 for a cartridge with 2K of memory, $30 for a 4K cartridge and $40 for an 8K cartridge. Most of the currently available cartridges are 2K programs. The price of the cartridges is about right and what one has come to expect. The $500-plus for the unit itself sounds a bit pricey to me although we'll see if there is any discounting once it hits the retail stores in mass. Umtech plans to distribute the VideoBrain through various department stores such as Macy's and the May Company primarily, although some retail computer stores and electronic outlets will undoubtedly carry it too. ■

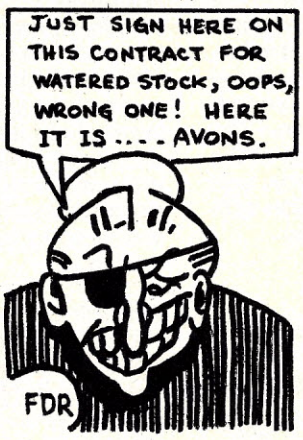


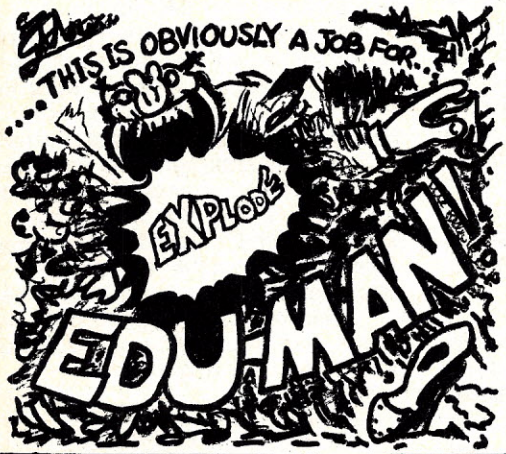
"Says 'How the #*??() should I know?'"

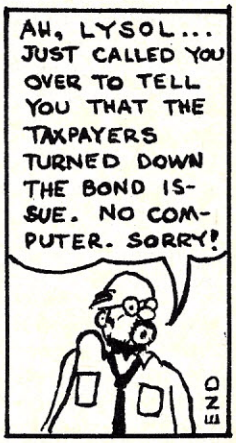
EDU-MAN MEETS THE AVON LADY!

©D.AHL

OUR STORY OPENS AS CREATIVE COMPUTING SUPER SALES REP, RONALD LYSOL, SITS WAITING IN THE OFFICE OF PRINCIPAL X WHO HAS EXCUSED HIMSELF TO GO TO THE W.C. (ASTUTE READERS WILL RECALL THAT MILD LYSOL IS, IN REALITY, NONE OTHER THAN EDUMAN.) THINKING, BY MISTAKE, THAT LYSOL IS H.S. PRINCIPAL X, THE AVON LADY ENTERS WITH HER BAG OF TRICKS.







WITH ALL DUE APOLOGIES TO OUR GOOD FRIENDS IN SOUTHBORO WHOSE NOVAS ARE RELIABLY SERVING HUNDREDS OF SCHOOLS AND COLLEGES TODAY.

JADE

FOR ALL CUSTOMERS EXCEPT CALIF. CALL TOLL FREE 800-421-5809

21L02 (450ns) Static Rams 100 @ $1.10 ea.	1702A E—PROM 8 @ $3.75 ea.	6502 Microprocessor 5 @ $11.00 ea.	2708 (450ns) E—PROM 8 @ $11.00 ea.	21L02 (250ns) Static Rams 100 @ $1.36 ea.	Z—80A Microprocessor 5 @ $25.00 ea.	8212 8 Bit I/O Port 25 @ $3.00 ea.	4116 (200ns) 16K Dyn. Ram 16 @ $24.00 ea.
Z-80 Microprocessor 5 @ $20.00 ea.	8224-4 Clk.Gen.&Dvr. 25 @ $8.75ea.	410D (200ns) Static Ram 100 @ $8.75ea.	4096 Dynamic Ram 100 @ $3.50 ea.	4200A (200ns) Static Rams 25 @ $10.00 ea.	74LS367 Hex Buffer 100 @ .70¢ ea.	74LS368 Hex Inverter 100 @ .70¢ ea.	2513 (5v) Character Gen. 5 @ $9.00 ea.

MICROCOMPUTER COMPONENTS

MICROPROCESSOR'S

F8	16.95
Z80	22.00
Z80A	28.00
CDP1802CD	19.95
2650	24.95
AM2901	22.95
6502	11.95
6800	18.95
6802	25.00
8008-1	9.95
8035	22.00
8080A	11.95
8085	27.00
TMS9900TL	75.00

8080A SUPPORT DEVICES

8212	3.50
8214	9.00
8216	3.75
8224	3.50
8224-4	9.95
8226	3.95
8228	7.95
8238	7.50
8251	9.95
8253	21.95
8255	21.95
8257	21.95
8259	21.95
8275	75.00
8279	20.00

FLOPPY DISC CONTROLLER

1771B	55.95
1771B-01	57.95

KEYBOARD CHIPS

AY5—2376	13.75
AY5—3600	13.75

PROM'S

1702A	4.00
2704	15.00
2708	12.00
2716	30.00
2716 Intl	38.00
2758	26.60
D3601	4.50
D3604	13.00
5203AQ	5.00
5204AQ	7.50
6834	17.50
6834-1	14.95
82S23B	4.00
82S129B	4.25
8223B	3.50

6800 SUPPORT

6810P	4.95
68B10P	6.00
6820P	7.50
6821P	7.50
6828P	11.25
6834P	16.95
6850P	9.75
6852P	11.75
6860P	10.00
6862P	14.50
6871P	28.00
6875P	8.75
6880P	2.50

Z80 SUPPORT DEVICES

3881	12.95
3882	12.95

STATIC RAMS

21L02	1.50	1.24	1.18
21L02 (350)	1.60	1.35	1.25
21L02 (250)	1.75	1.60	1.50
410D	10.75	10.00	9.25
1101A	1.00	.90	.80
2101-1	2.95	2.75	2.60
2102	1.25	1.15	1.00
2111-1	3.95	3.50	3.25
2112-1	2.95	2.80	2.69
2114-3	11.00	10.00	9.25
2125L	11.10	9.00	8.30
2147	37.50		
31L01	2.50	2.35	2.00
3106	3.95	3.70	3.25
3107	3.95	3.70	3.25
TMS4044	9.95	9.00	8.95
4200A	12.95		
TMS4045	11.00	10.00	9.25
5101	8.30	7.40	7.25
74C89	3.25	3.05	2.85
7489	2.25	2.10	1.90
74S201	4.50	4.00	3.75
P8101	4.20	3.40	2.80
P8155	17.00	14.00	
P8156	21.00	18.00	
8599	1.88	1.75	1.60
9102BPC	1.65	1.45	1.30

CHARACTER GENERATORS

2513	6.75
2513 5v upper	9.75
2513 5v lower	10.95
2516	10.95
MCM6571	10.95
MCM6571A	10.95
MCM6574	13.25
MCM6575	13.25

WAVEFORM GENERATOR

8038	3.50
MC4024	2.25
566	1.50

DYNAMIC RAMS

416D/4116	32.00
1103	1.00
2104	4.00
2107B	4.25
2107B-4	3.95
TMS4050	4.00
TMS4060	4.50
TMS4070-2	32.00
4096	4.00
4116/416D	32.00
MM5270	4.50
MCM6605	5.00

USRT

S2350	10.75

UART'S

AY5-1013A	5.25
AY5-1014A	8.25
TR1602B	5.25
TMS6011	5.95
IM6402	10.80
IM6403	10.80

MISC. OTHER COMPONENTS

N8T20	3.39
N8T26	2.10
N8T95	1.35
N8T96	1.35
N8T97	1.35
N8T98	1.35
81LS95	2.00
81LS97	2.00
1488	1.75
1489	1.75
D3205	4.00
D3207A	4.55
D3208A	14.20
D3211	10.00
B3222	9.75
B3242	10.15
D3245	5.60
C3404	6.75
P3408A	12.00
P4201A	5.20
MM5320	7.50
MM5369	1.90
TMS5501	24.95
DM8130	2.90
DM8131	2.75
DM8833	2.50
DM8835	2.50
DM8837	1.75
MK50240	20.00
MK50250	15.00

JADE 8080A KIT

$100.00 KIT

BARE BOARD $30.00

E—PROM BOARDS

MR-8 (8K uses 2708) with 1K RAM	KIT	$99.50
MR 16T (16K uses 2716) with 1K RAM	KIT	$99.50
MM-16 (16K uses 2708) RAM/N/ROM (16K uses any E-PROM)	KIT	$99.00
JG-8/16 (uses 2708 or 2716)	KIT	$117.00
BARE BOARD		$59.95 $30.00

TV-1

Convert T.V. set to Video Monitor.

KIT $8.95

STATIC RAM BOARDS

8K

250ns	ASSEMBLED & TESTED	$189.95
450ns	ASSEMBLED & TESTED	$149.75
250ns	KIT	$169.95
450ns	KIT	$125.00
BARE BOARD		25.00
6800	ADAPTER to S-100 System KIT	$12.95

16K

250ns	ASSEMBLED & TESTED	$435.00
450ns	ASSEMBLED & TESTED	$380.00
450ns	KIT	$335.00

32K

250ns	ASSEMBLED & TESTED	$850.00
450ns	ASSEMBLED & TESTED	$775.00
450ns	KIT	$675.00

DYNAMIC RAM BOARDS

On board Refresh power is provided with no wait states or cycle stealing required.
+8VDC 400MA DC, +18VDC 400MA DC and −18VDC 30MA DC.

EXPANDABLE 32K

8K (375ns)	KIT	$151.00
16K (375ns)	KIT	$259.00
24K (375ns)	KIT	$367.00
32K (375ns)	KIT	$425.00

EXPANDABLE 64K

16K (375ns)	KIT	$281.00
32K (375ns)	KIT	$519.00
48K (375ns)	KIT	$757.00
64K (375ns)	KIT	$995.00

MOTHER BOARD's — S-100 Style

13 slot — w/front panel slot

BARE BOARD $35.00

KIT $95.00

22 slot — $149.95

ASSEMBLED & TESTED

THE PROM SETTER

WRITE & READ EPROM

1702A — 2708 — 2716

5204 — 6834

- Plugs directly into your ALTAIR IMSAI Computer
- Includes Main Module Board and External EPROM Socket Unit.
- The EPROM Socket Unit is connected to the Computer through a 25 pin connector.
- Programming is accomplished by the Computer.
- Just read in the Program to be Written on the EPROM into your Processor and let the Computer do the rest.
- Use Socket Unit to Read EPROM's Contents into your Computer.
- Software included.
- No external power supplies. Your computer does it all.
- Doubles as an Eight Bit Parallel I/O.
- Manual included.

KIT $210.00

ASSEMBLED $375.00

KIM—1

ASSEMBLED & TESTED $245.00

MEMORY PLUS

for KIM—1

8K RAM (21L02)

8K EPROM

ASSEMBLED & TESTED

$245.00

JADE Z80 KIT

—with PROVISIONS for ONBOARD 2708 and POWER ON JUMP

$135.00 EA. (2MHZ)

$149.95 EA. (4MHZ)

BARE BOARD $35.00

JADE Z80 "UPGRADE" KIT

Change your JADE 2MHz Z80 to a 4MHz version with this simple kit:

only $17.95 with trade

$49.95 purchase.

To trade, you must give us your 2MHz Z80 chip and 8224 clock driver.

The "UPGRADE KIT" includes:

Z80A chip,	1.8K resistor
8224-4 clock driver	20 pf. capacitor
36MHz crystal	

COMPU/TIME CT 100

S100 BUS COMPATIBLE

TIME & CALENDAR

Microprocessors need the power that a real-time clock can offer. Date and time becomes instantly available. **COMPU/TIME** does not have to be initialized every time the system is powered up. It possesses a crystal-controlled time base to obtain superior accuracy and has two setable coincidence counters. Time, date, and counters are set via software.

COMPUTATIONAL FUNCTION

Microprocessors need to be complimented by hardware arithmetics to free up memory pages dedicated to floating point routines and mathmatical software. **COMPU/TIME** provides a 40 function calculator array so that algebraic, trigonometric, basic arithmetic problems can be solved without the need of developing sophisticated software.

Buy It Your Way

COMPU/TIME	CT100	$199	KIT	$245	Assembled
COMPU only	C101	$149	KIT	$189	Assembled
TIME only	T101	$165	KIT	$205	Assembled
COMPU/TIME	PC Board only		$80		

JADE VIDEO INTERFACE KIT

FEATURES $99.95

- S-100 Bus Compatible
- 32 or 64 Characters per line
- 16 lines
- Graphics (128 x 48 matrix)
- Parallel & Compositive video
- On board low-power memory

Powerful software included for cursor, home, EOL, Scroll Graphics/Character, etc.

Upper case, lower case & Greek

Black-on-white & white-on-black

full ASCII PROFESSIONAL KEYBOARDS

- Full 128 Character ASCII!
- Tri-Mode MOS Encoding!
- MOS DTL TTL Compatable Output!
- Two-key Rollover!
- Level and Pulse Strobe!
- Shift and Alpha Lock!
- Selectable Parity!
- Positive or Negative Logic!

MODEL 756 (56 keys)

PRICING INFORMATION

Model 756 (assembled)	$75.95
Model 756K (kit)	64.95
Model 702 Enclosure	29.95
Model 710 Numeric Pad	9.95
Model 756MF Mtg. Frame	8.95

CONNECTORS

DB — 25P	$3.00
DB — 25S	$4.00
COVER	$1.50
44 Pin — PC & EYE	$1.95
44 Pin — WW	$2.50
86 Pin — (6800) PC	$5.00
86 Pin — (COSMAC ELF) PC	$5.00
100 Pin — (Altair) PC	$4.50
100 Pin — (Imsai) PC	$3.75
100 Pin — (Imsai) WW	$4.25

JADE

REAL TIME CLOCK FOR S-100 BUS

1 MHZ Crystal Oscillator
Two independent interrupts
One interrupt uses 16 bit counter in 10 USEC steps
Other interrupt is in decade steps from 100 USEC to 10 sec.
Both software programmable
Board can be selected by 128 device code pairs.
Complete documentation includes software to display time of day.
Double sided solder mask
Silk screen parts layout

JG—RT ASSEMBLED & TESTED $179.95

JG—RT KIT $124.95

BARE BOARD with Manual $30.00

TARBELL CASSETTE INTERFACE

- Plugs directly into your IMSAI or ALTAIR*
- Fastest transfer rate: 187 (standard) to 540 bytes/second
- Extremely Reliable—Phase encoded (self-clocking)
- 4 Extra Status Lines, 4 Extra Control Lines
- 37-page manual included
- Device Code Selectable by DIP-switch
- Capable of Generating Kansas City tapes also
- No modification required on audio cassette recorder

JADE KIT* $99.95 ASSEMBLED $175.00

*(6 month warranty from JADE) MANUAL $4.00

JADE PARALLEL/SERIAL INTERFACE KIT

$124.95 KIT

S-100
2 Serial Interfaces with RS232 interfaces or 1 Kansas City cassette interface.
Serial interfaces are crystal controlled.
Selectable baud rates.
Cassette works up to 1200 baud.
1 parallel port.

DATA COMMUNICATIONS ADAPTER

80-103A Serial I/O and FSK modem for professional and hobby communications.

- Completely compatible with your IMSAI, ALTAIR* SOL** or other S-100 microcomputers. Trademarks of *MITS, **Processor Technology
- Designed for use on the dial telephone or TWX networks, or 2-wire dedicated lines, meets all FCC regulations when used with a CBT coupler.
- All digital modulation and demodulation with on board cyrstal clock and precision filter mean that NO ADJUSTMENTS ARE REQUIRED
  - Bell 103 standard frequencies
  - Automated dial (pulsed) and answer
  - Originate and answer mode
  - 110 or 300 BPS speed select
  - Complete self test capability
  - Character length, stop bit, and parity
  - 90 day warranty and full documentation

PRICES: BARE BOARD and Manual $49.95

Assembled (48 hr. burn in) $279.95

JG—DCA KIT $159.95

NUMBER CRUNCHER

The CT200 is a number-oriented microprocessor intended for use in those applications that require fast versatile mathematical solutions.
THIS IS NOT A CALCULATOR CHIP. THERE ARE NO KEY DELAYS.
The CT200 has a unique architecture that is designed to be a TASK processing system within a system. This unique architecture will allow the CT200 to work and run with ANY S100 BUS microprocessor system. It is completely compatible with Z80, 4MHZ version also, 8080, 6800, 6502 microprocessor. A micro incoded instruction set allows programming in a calculator like language. The instruction set includes a full set of test and branch instructions. All decoding of S100 bus signals (for select or control functions) is performed with strobed latches to eliminate the possibility of glitches.

PRICE: $249.00

Includes - Manual, ASSEMBLED & TESTED.

JADE

Computer Products

BANKAMERICARD VISA

5351 West 144th Street
LAWNDALE, CALIFORNIA 90260
(213) 679-3313

RETAIL STORE HOURS: Monday - Friday 9—7
Saturday 9—5

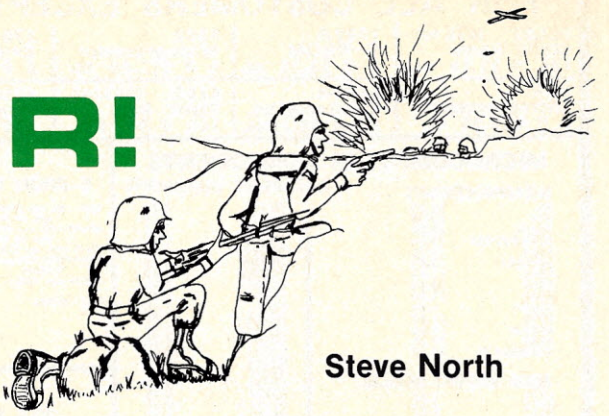
Discounts available at OEM quantities ADD $1.50 under 10 lbs. for shipping. California residents add 6% sales tax.

NEW CATALOG NOW AVAILABLE

CIRCLE 112 ON READER SERVICE CARD

A Creative Computing Software Profile...

ENCOUNTER!



A real-time combat game, each player moving independently, typing as fast as he can, with minimum time to consider his moves.

Steve North

It is estimated that 75% of all personal computer systems are used solely for games and recreation. Consequently there is quite a bit of interest in games for personal computers, especially those featuring unique graphic displays. I recently had the chance to try out one such game, called ENCOUNTER! written and marketed by Objective Design, Inc.

Encounter! is a game written in 8080 assembly language. An 8080-based microcomputer, two keyboards, and a memory mapped video display with video inversion (such as the Processor Tech VDM-1) are required to play the game. Encounter! is a board game played in realtime. Moves are accepted from the players independently, as fast as they can be typed. So Encounter! is an entirely different type of game than Chess, where moves are made alternately. There is no time for lengthy consideration of your moves.

The board in Encounter! is divided into districts, referred to by column number (A-J) and row number (1-12). A district may be empty, occupied, or blocked. Empty districts are represented by spaces, and occupied districts by the number of men in the district (flanked by either +s or -s to indicate which side occupies the district—as in +23+ or -05-). Blocked districts, which serve as obstacles in the game, are shaded in. At the outset of the game, each player sets up his portion of the board, while the other player leaves. Thus the players do not know the other player's initial setup until the game actually begins. When both players have finished the set-up procedure of distributing a limited number of men within his quadrant of the board, the game is started.

Commands

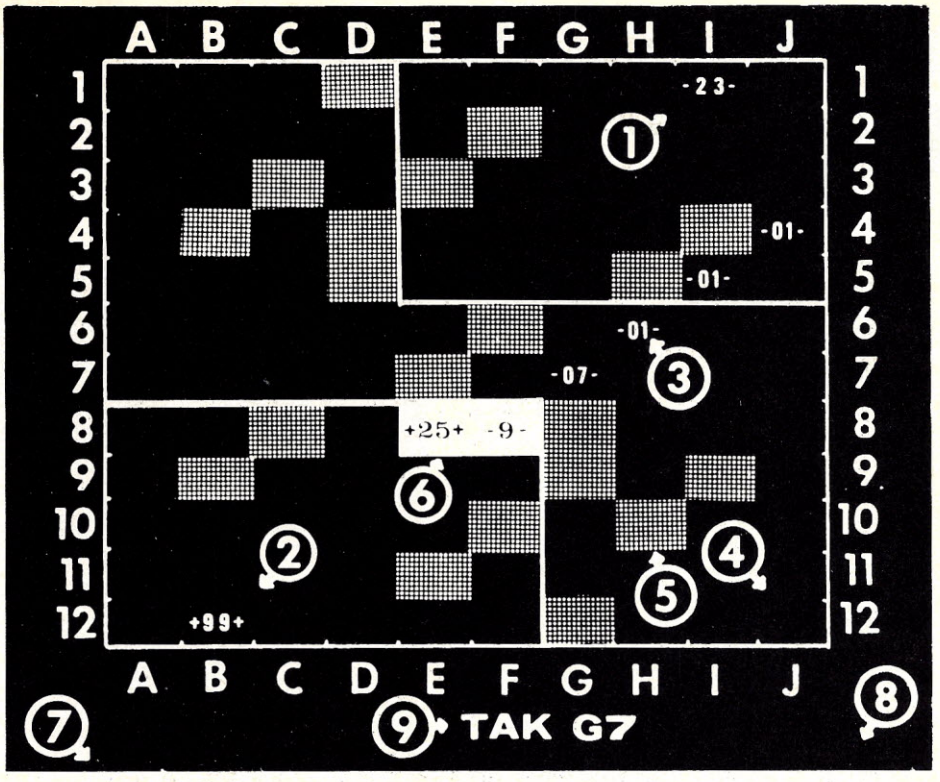
The object of the game is to wipe out all the opponent's men, or his home district, depending on the particular version of Encounter! played. Each side has one district designated as the home district. Side 0's home district is located in the lower lefthand side of the board, and side 1's home district is in the upper righthand corner. During execution of the game, you have at your disposal the commands shown in the box.

The Display

While the game is in progress, various portions of the screen indicate what the game is doing. At the bottom left side of the screen, player 0's input is echoed, and at the bottom right, player 1's. This actually permits you to observe your opponent's commands, though you rarely have time to do so in actual play. If a command is illegal, the word REJECT appears next to it. The line immediately above the command entry line is used to display informatory messages for each side. These messages tell you if you've won or lost a district, or if there is an immediate loss on your side, or if you've broken off an attack. The area above this is used for the board. The border of the board

COMMANDS FOR ENCOUNTER!

Command	*Example*	*Description*
M	M 10 A12 C9	Move men from one occupied district to another. In the example, we moved 10 men from A12 to C9.
T	T 10 E8 F8	Transfer men from an occupied district to an adjacent unoccupied district. This command is used to expand into empty areas on the board. In the example we transferred 10 men from E8 to F8.
A	A A12 B12 A A12 B12 L5 A A12 B12 X6 A A12 B12 L5 X6	Attack an adjacent enemy district. When this command is executed, the computer automatically controls the encounter between the two opposing districts, displaying the decreasing number of men within both districts. There are two optional parameters that can be set in this command. When you win an encounter, men from the attacking district are automatically transferred into the taken district. The L parameter specifies how many men to leave in the attacking district. There is also an automatic breakoff point, so that an attacking district can not attack itself out of existence. When the number of men reaches below this level (the X parameter), the attacking district breaks off the attack automatically. The breakoff point must always be higher than the number of men to leave, and at least one man must be left.
X	X E8	Break off attack.
D	D L5 X6	Sets default values for parameters in the Attack command.



The circled numbers indicate game features: (1) side 1 home district; the exact location is programmable; (2) side 0 home district; (3) occupied district; (4) unoccupied district; (5) blocked district; (6) attack in progress; reverse video; (7) keyboard 0 command entry display area; typical command: M 33 0 E8; (8) keyboard 1 command entry display area; typical command: T 20 I1 I2 I3 I4 REJECT; (9) field messages line. The dark lines show the setup limits of both sides, and do not actually appear on the screen. The limits are programmable for each game.

displays the coordinate system, while the rest of the board is a display of the playing area. Areas engaged in encounters are displayed in reverse video (white on black, rather than black on white). There are also two displays of the game timer, located on the bottom left and right sides of the screen.

During the game, men are naturally lost due to attacks. However, each side has a birthrate, determined by the number of districts that side occupies. Births occur in the home district only, so it is important to protect the home district. Encounter! comes configured as three different games. The rules are basically the same for all the games, but the variations are in the blocked districts, the number of men each side starts with, and the conditions for a win.

When the game is started, the version of Encounter! to be played and the level of play (speed of the game) are selected. There is a time delay in the execution of most of the commands (such as Transfer) which makes it possible for a slow typist or inexperienced player to use the game. So the ability to set the speed of the game is quite desirable, since it makes the game playable for both beginners and experts. (A member of the staff at *Creative* recently remarked that game writers sometimes play their games so much that they become experts and make them trickier, so that a beginning player may find many games bewildering.)

Playing Encounter!

A game like Encounter! permits many different styles of play and strategy within a simple framework of rules. For instance, the manual suggests starting an attack at one area of the board with a high breakoff point, and while the opponent handles this problem, starting another elsewhere. Also, since it is to your advantage to spread out a bit, so as to obtain a high birthrate, it is sometimes possible for an opponent to break through into a sparsely populated area and to move quickly towards your home district. You can try keeping the other player busy in one portion of the board while you sneak over into another part. Really, it is necessary to play Encounter! for a while to appreciate all the techniques you can try out. In some ways it is like the games of Risk and Chess.

Documentation

The Encounter! documentation is generally excellent. The instructions include lots of explanation, examples, and a diagram or two. The style of the manual is quite clear and interesting. The source code for the game is provided, but is unassembled. It is hard to understand what good this really does for someone who wants to modify the game, unless they want to type in thousands of characters of source code. However some details on modification of the game (for custom versions of Encounter!), with references to the appropriate memory locations, are given.

One big problem with Encounter! is that the object code is assembled to run at 8000 hex (starting at 32K). Very few people have memory up there. A much more reasonable place would have been 0000. On the other hand, it isn't going to kill anyone to pull out a memory board and set the board address DIP switch. Larry Weinstein of Objective Design explained that the rationale for providing the object code for that address was not the probable location of RAM but the probable location on non-writable ROM. Larry said that they could have provided the code at 8K or 16K, but even then there are people who don't have that much memory. I still don't see why they couldn't have provided it at 0000 (the only computers I can think of offhand that have ROM down at 0000 couldn't run Encounter! anyway) but as Larry said, it's no big deal. You will need two terminals or keyboards to play Encounter!, but that goes without saying.

Larry Weinstein also mentioned that he was surprised that more people aren't writing sophisticated video games for their systems, but are instead content to play Bagels for the two-hundredth time. One of the reasons for this may be that games like Encounter! or TREK-80 or Spacewar (see the July-August issue of *Creative*) have to be programmed in assembly language. A BASIC interpreter is just too slow and clumsy for writing complex, high-speed video games. If you don't believe it, write a routine in BASIC to clear the VDM screen using the POKE statement. You will of course have to convert the memory addresses from hex to decimal, etc. Then see how long (yawn) it takes to clear the screen. So because of a lack of compilers with special graphics features, video games must be written in assembly language or perhaps BASIC with machine-language subroutine calls. Since most people find programming in assembly language tedious and error-prone, as well as more difficult than programming in a high-level language, it is a lot easier to just play Bagels (again).

At any rate, if you're interested in video games, and especially in two-player games, this is a product that will provide you and someone else with many hours of fun. Encounter! is available on papertape for $16.95, or Tarbell cassette for $19.95, postpaid, from Objective Design, Inc., P.O. Box 20325, Tallahassee, Florida 32304. ■

Can a Computer Really Play Winning Chess?

David H. Ahl

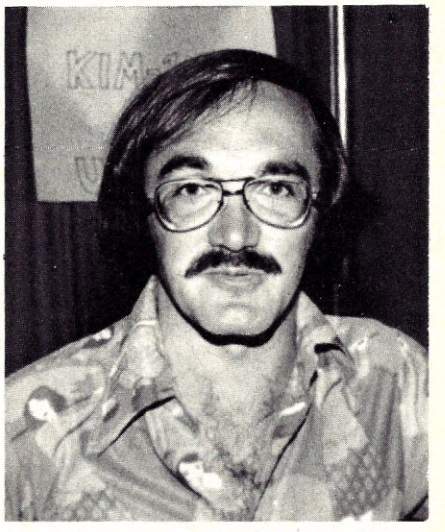
Back in the '50s when Elvis was making like a hound dog, *Victory At Sea* was NBC's top prime-time hit, and Shockley demonstrated the first transistor at Bell Labs, some optimistic researchers speculated that computers really ought to be able to think. At that time, it was generally accepted that one good measure of "thinking" was playing a good game of chess. I'm sure that many chess experts and amateurs wouldn't disagree with that view today.

Chess is the intellectual game *par excellence.* There is no random chance involved, just sheer intellect in a situation so complex that neither player can hope to understand it completely, but sufficiently structured that each can hope to outthink the other. Even after 200 years of exhaustive play and thorough analysis, the field is still ripe for further exploration and development. Thus, if one could devise a successful chess program, one would seem to have penetrated to the core of human intellectual endeavor.

Three researchers in particular delved into this problem with great fervor. They were, of course, Allen Newell and J. C. Shaw at Rand Corporation, along with Herbert Simon at Carnegie Institute of Technology, now Carnegie-Mellon University.

During their efforts to produce a good chess-playing program, they discovered a number of things. First of all, it's not easy. Indeed, it may not be possible at all to produce a program that can play at the master level. Why not? First (and the reason that chess-playing programs are so good as a measure of thinking) is that the possible number of moves is in the neighborhood of infinity, give or take a bit. (Actually, Claude Shannon estimated that there are something like 10^{120} possibilities, which doesn't help us too much, since there are only 10^{16} microseconds in a century!). Consequently, you can't store all the possibilities in memory of any kind, nor could you possibly analyze them all. The approach, therefore, is to "teach" the computer how to play chess, evaluate possible moves, and formulate a playing strategy.

To teach a computer, it is probably helpful to first evaluate how a human plays chess. Most players have either learned from experience, or been taught by another player or book, that beginning game moves are best made following proven approaches. These openings minimize one's vulnerability and hopefully create a strong position for the attack to be launched in the middle part of the game. These "relatively" few openings, about 20 or so, follow reasonably predictable courses of action for perhaps the first seven to ten moves. Thus the opening of the game can largely be played from memory — human or computer.

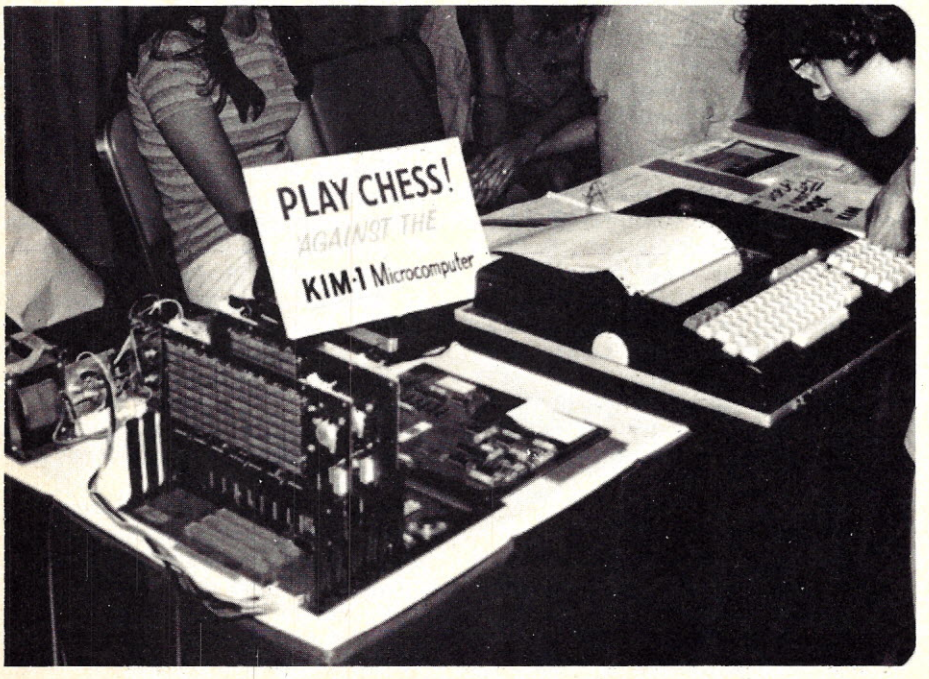


Peter Jennings, president of Micro-Ware Limited, is the co-creator of Microchess.

The middle part of the game gets a bit hairy. Moves aren't nearly as predictable and most players adopt a strategy of evaluation of each move by considering its consequences after several more moves on both sides. This is, of course, what separates the men from the boys, so to speak, with the expert player able to accurately evaluate the consequences of a move 12 to 15 moves later while most novices are able to look ahead only 2 or 3 moves.

Different players put different importance on various pieces and positions. However, there seems to be general agreement that at least five factors should be considered in evaluating a move: mobility, value of attacked piece, vulnerability (and value) of attacking piece, King safety, and overall board configuration. Clearly, you don't just "tell" the computer these things and then say, "OK, now play."

If the middle game is hairy, the end game is downright mind-boggling when each side has maybe six or seven pieces left. Sides of the board have largely lost their meaning and configurations are possible that may *never* have occurred before. A strong attack can lead to an impossible defense in the span of just eight or nine moves. In short, heuristics and textbook approaches begin to break down and each player tends to develop his or her



Microchess on the KIM-1 proves a formidable opponent to the unwary at various computer conferences (here at PC'77 in Atlantic City).

own individual strategy. What does one do with a computer program at this point? Good question, with as many answers as there are programmers.

The ACM and some other groups have been running computer-chess tournaments for years. David Levy, an international master, regularly beats the winning computer at the end, although the playing is getting awfully good. Many of these programs are written on big (read, gigantic) machines of the CDC 6600 class, although some are on minis (I recall a Nova that played extremely well in the 1975 tournament in Minneapolis).

If the middle game is hairy, the end game is downright mind-boggling when each side has maybe six or seven pieces left.

Given this background and the difficulty of the problem, what can we expect from a micro? A year ago, I would have been tempted to say, "Not much." But Peter Jennings has proven that statement quite wrong.

Peter wrote a program called Microchess for the KIM 1. After all, what else can you do with a bare board, not-very-expandable microcomputer with only 1K of memory? Only 1K — you must be kidding! But I'm not kidding. Of course, it has a rather powerful 6502 chip at the heart of the system.

Peter follows the general outline above. The opening game consists of nine moves on both sides, in which the computer plays from a table if possible. This table occupies about 200-300 bytes. From there on, the program looks ahead approximately three moves on both sides and evaluates the possible outcomes from the standpoint of mobility, value of piece under attack and value of pieces open to attack. The program only uses castling in the opening and cannot capture pawns *en passant* — minor limitations in my opinion.

Peter has recently converted the program to run on 8080 systems; however, it requires 4K of memory. Actually, 2K is for I/O so the equivalent 1K 6502 program occupies 2K on an 8080 system. Interesting.

The accompanying manual is quite thorough and includes a description of the program, a sample run, and appendix with details of converting the I/O routines to your particular system. Microchess is available on cassette for the SOL or on paper tape for other systems. An interesting feature is the ability to reverse the board at any time which, if done continuously, lets the computer play against itself.

How good is Microchess? Well it beat me, but I'm a rank beginner, so that's no measure. Against Dark Horse, the program which won 6th place in the last ACM computer chess championship, Microchess was up one piece after 26 moves, but faltered and lost in the end game. (Microchess does not have a separate end-game strategy.) I had hoped to have the results of the game between Microchess and Fidelity Electronic's Chess Challenger, but that will have to wait until later.

Microchess costs $13 for the KIM-1, $18 for 8080 systems on paper tape or $20 on SOLOS cassette. Contact Peter R. Jennings, Micro-Ware Ltd., 27 Firstbrooke Road, Toronto, Ontario, M4E 2L2, Canada. (416) 424-1413. ■

Another View of Microchess

A most interesting thing about Microchess is the manner in which it generates moves. Unlike most larger chess-playing programs, Microchess selects a move as a result of a sequential search through all possible moves. This very primitive algorithm for move evaluation does not hamper Microchess as much as one might guess. An example of typical play by Microchess may be seen in the following opening game:

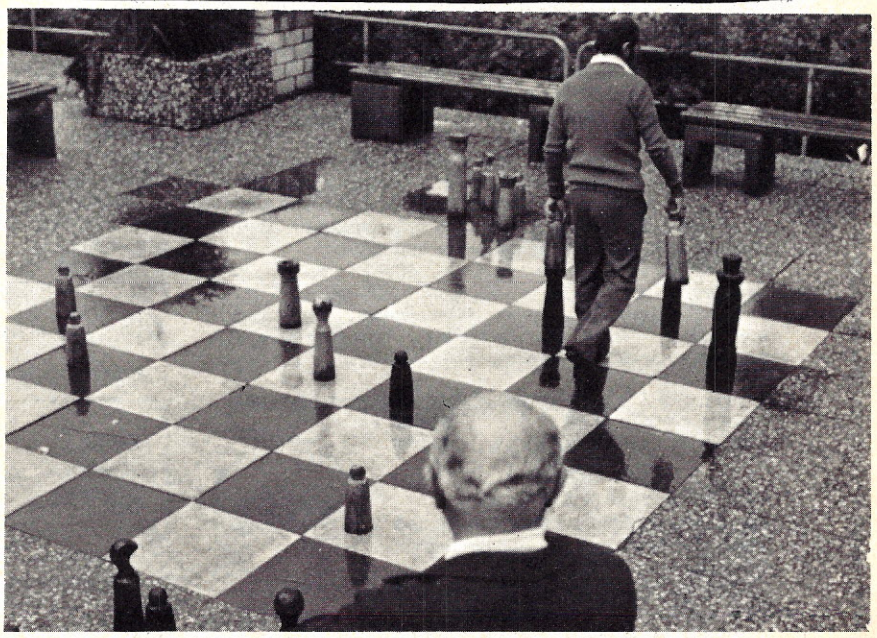
Microchess (white)	*Challenger* (black)
KP - K4	KP - K4
KN - KB3	KN - KB3
KN x KP	KN x KP
Q - KN4	KN - QB4
QP - QP4	Q - KB3
QP x KN	Q x KN
K - Q2	Q x KP
Q - K4	Q - K2
Q - Q4	

Obviously neither player was playing particularly inspired chess; however, all the moves made by Microchess were reasonable if not optimal. Unfortunately Microchess does not fare so well when the game requires a move that is not obvious in the current board position as can be seen by the following opening:

Challenger (white)	*Microchess* (black)
KP - K4	KP - K4
Q - KB3	Q - KR5
KB - QB4	QP - Q4
KB x QP	QBP - QB3
Q x KBP	K - Q1
Q x KB	K - QB2
Q x KNP	KN - K2
Q x KP	K - Q1
Q x KR	

Another flaw in this program is the fact that the internal board representation used by Microchess will allow only one Queen per side at any time. Thus the value of pawn promotion is considerably reduced.

Richard Freeman
Irvine, CA



Macrochess is a popular way to spend an afternoon in Frankfurt-am-Main. While games are played between individual players, each has a team of advisors (hecklers?) ready with opinions and recommendations.

The Music Cassette

David H. Ahl

We're always glad to see second sources of hardware and boards for computer systems. It's good for the consumer. Generally the second source offers advances in technology over the original manufacturer, lower prices, or both. The manufacturer then responds with a second-generation product and the whole market benefits. The same is true, of course, with respect to software. The more companies marketing software for a computer system, the better. Not only will the user have more choice, but the hardware manufacturer will be able to concentrate on state-of-the-art hardware advances while the software company focuses on the applications.

The Music Cassette by HUH is the third step in the chain. Namely, a company marketing applications software (actually data) that runs on a second-source software system (the Music System by Software Technology) designed for a hardware system (the SOL-20 by Processor Technology).

HUH's Music Cassette #1 contains no actual systems software, but rather the data for seven musical selections to be played under the Software Technology Music System. It's easy to use. First, load the Music System on the SOL by simply typing GET and, after the tape has read, type EX O. Type an R to get back to SOLOS (or CUTER). Then swap tapes and type GET again to get the first selection off the Music Cassette. When it has read, type EX O to get back to the Music System. Then simply follow the steps in The Music System manual — F to initialize the file, S to create the machine executable score and P to play the composition.

All the arrangements on The Music Cassette are by Alan Rawson. The seven compositions are:

1. *Star Wars Main Title Theme,* by John Williams.
2. *Invention #8,* from the 2 and 3 part Inventions by J.S. Bach
3. *Flight of the Bumblebee,* by N. Rimsky-Korsakov
4. *Boogie Woogie, by Pinetop Smith*
5. *The Easy Winners,* by Scott Joplin
6. *Fugue #7,* from the Well-Tempered Clavier by J.S. Bach
7. *Minute Waltz,* by Frederic Chopin

We frequently play our SOL at trade shows and some of our booth personnel have complained that eight hours a day for three days of nothing but the baroque music supplied with the Software Technology Music System is a bit too much. (Linda Harrison feels that's putting it much too mildly). Anyway, it's nice to find a tape like this if you don't have the time or inclination to transcribe a body of music yourself. Mark Garetz of HUH mentioned to me at the West Coast Computer Faire that they are paying 2.75¢ to Star Wars per cassette royalty. This, of course, is the reason that you don't see more tapes of current and popular music commercially available.

The tape is produced by HUH Electronic Music Productions. We're not sure what HUH stands for, if anything (Horrendous Unusual Heuristic leaps to mind but...). The tape costs $19.95 from HUH, P.O. Box 259, Fairfax, CA 94930. (415) 457-7598. ■

Seven Super Computer Stores

BYTE SHOP 3 OF SAN JOSE
COMPUTER STORE
(唯一中國人開設的電腦零售店)

你想自己裝配一個小型電腦嗎？

請到我的展覽室來 (2626 Union Ave, S. J. Ca 95124) 實際操作你所需要的電腦、計算機等並讓我解答你的一切問題，幫助你按裝或者你要我替你設計裝配請電話408-377-4685找RAY LYN (林) (OWNER) 並零售所有一切零件，以及 Hardware. Software For Micro Computers.
開課教授 (晚上) (Offer Classes)
請剪下此廣告，學費便宜 5 %

CIRCLE 132 ON READER SERVICE CARD

Computer Mart New York

New York City & Long Island

IMSAI, SWTPCo, Digital Group
Processor Tech, Apple, OSI
TDL-Z-80, Seals, Cromemco,
Veras, Tarbell, Oliver

Magazines, books, chips,
sockets, connectors, terminals.

IT'S ALL HERE WAITING FOR YOU
FRIENDLY ADVICE TOO

New York City
118 Madison Ave.
(Enter on 30th St.)
New York 10016
212-686-7923
(Tues. thru Sat.)

CIRCLE 126 ON READER SERVICE CARD

THE COMPUTER HARDWARE STORE INC.

Dealers for:
APPLE II, IMSAI
VECTOR GRAPHIC, KIM-1
TECHNICO, OAE
CYBERNEX, JIM-PAK
TERMINALS, PRINTERS
BUSINESS SYSTEMS, BOOKS
SOFTWARE AND MUCH MORE

CATALOGUE AVAILABLE

818 Franklin Street
Alexandria, Virginia
(703) 548-8085

9 West Cary Street
Richmond, Virginia
(804) 780-0348

CIRCLE 136 ON READER SERVICE CARD

THE ELECTRONICS PLACE

* * * Vector Graphics
* * * S W T P C
* * * Cromemco
* * * Kim-1
* * * North Star
* * * Tarbell

Sales & Service, Magazines & Books

7250 McKnight Road
Pittsburgh, Pa. 15237
(412) 367-2900

CIRCLE 146 ON READER SERVICE CARD

Personal Computer Corporation

We know EDUCATION
We know COMPUTERS
We have, on the premises

- FULL TIME SERVICE & REPAIR
- FULL TIME PROGRAMMING

We Accept:
Master Charge
BankAmericard
Purchase Orders

ASK FOR: EVERETT
DAVE
ED

FOR ALL
YOUR MICROCOMPUTER
REQUIREMENTS

Frazer Mall, Rtes. 30 & 352
Malvern, PA 19355
Phone: (215) 647-8463

CIRCLE 143 ON READER SERVICE CARD

VIRGINIA HOME COMPUTER CENTER

DEC
Apple
Vector Graphics
TDL
Polymorphic
RCA
Processor Technology
North Star
Digital Systems
Persci

2927 Va. Beach Blvd.
Va. Beach, VA 23452
(804) 340-1977

12588 Warwick Blvd.
Newport News, VA 23606
(804) 595-1955

CIRCLE 140 ON READER SERVICE CARD

COMPUTERS PLUS INC.

678 S. Pickett St.
Alexandria, VA 22304

Ask for Bob or Dan.
Northern Virginia's
Newest and Finest
Microcomputer Store

(703) 751-5656

CIRCLE 138 ON READER SERVICE CARD

If It's Right, You Know It

Debbie Schulz

So you're thinking about a career in Data Processing? That could just be one of the worst decisions you'll ever make. Take my very good friend, Janice, who's in programming school. Now Jan's not a dumb girl and she tries very hard, but Data Processing and her just DO NOT get along.

But then, deciding to go into Data Processing could just be one of the best choices you'll ever make. Several other friends, Joyce, Cathy and Nancy, are each in separate fields of Data Processing — Joyce is a computer operator, Cathy's a programmer and Nancy does system analysis work. Each loves her job and can't imagine doing anything else.

Joyce, a graduate of a local computer college, admits school isn't everything. "In fact, you don't really need any training to get a job as a computer operator. All you need is the desire to learn."

Joyce's first job was on the third shift at Little Brown Shoe Company, making only $7,800 a year. "But," she contends, "this job taught me most of what I know. I learned how to run all the machines and produce records. I also learned how to keep maintenance reports on errors and helped devise a new system for early detection of errors."

Today, Joyce is Manager of Computer Operations at Chemicals Diversified, making a respectable $9,900 a year. Her duties include planning, organizing, and controlling computer operations. She also establishes different schedules for the use of equipment, and, as head supervisor, manages three people.

I asked Joyce what she likes best about her job. "The fact that it's never boring — there's always something new and challenging happening. Like yesterday: one of our new card readers broke down — a card was stuck and jammed the machine. The guys I work with couldn't find anything to release the mechanism until I handed them a hairpin. It was just what they needed."

Joyce admits there may come a time when she'll get tired of computer operations and want something more. "If and when that time comes I'll probably go back to school. I think I'd like to try my hand at programming...."

Cathy, my programming friend, just graduated from a four-year college this spring. She works for the Kleasy Company, a business which specializes in food wholesaling. Her salary is around $10,000 a year. She explains her job this way. "A computer is a machine — and a dumb one at that. It has to be told when and how to do something, and, if you'll pardon the expression — where to go."

I asked her how she programs a computer to do something. "Well, first, to work with a computer you need to learn its languages — A/L, Cobol, Fortran; it's very much like learning French, Spanish or Latin. A/L means Assembler Language and is the basis for all other languages. Cobol means COmmon Business Orientated Language and Fortran, FORmula TRANslation, deals with math and chemistry."

She told me she's currently working on a project to increase the efficiency of Kleasy's payroll system. "Programming is problem-solving — applying the computer's time-saving speed and accuracy to solving many repetitious jobs."

It's obvious in talking to Cathy that she likes her job. She agrees with Joyce that her job is never dull, but she says, at times it's very frustrating. "There are days I just want to pack it up and leave. I guess I get frustrated too easily when things don't work exactly right — when I get an error in my program and can't find it or I get a problem I can't solve. Fortunately, the people around here are really great about helping with things like that."

Cathy's future looks bright; she hopes one day to be Manager of Data Processing at Kleasy's or to become a systems analyst.

The one who should think twice about programming is Janice, a student at a two-year technical college. She hates programming and as a result has no patience in solving her problems but instead goes to someone else for help. She says, "I do dumb things. I can't help it, they just happen. I've been working on a problem — had it all written in A/L and keypunched onto cards. Yesterday I dropped all my cards. I'm still trying to get them back in the right sequence."

Incidentally, Janice is three weeks behind the rest of the class in getting her lab problem done.

Data Processing is a career that, like most careers, takes dedication and a desire to learn. Data Processing teaches you to think and to think logically. To ask yourself, "What would happen if...", to analyze situations, their cause and effect, and to determine what you can do to improve them.

One person made for this kind of job is Nancy, a systems analyst with First County Bank. She started working with First County Bank in 1970, after graduating from college with a Master's Degree.

About her job and how far she's come in the last six years, Nancy says, "I started out with analyses of a less complex nature; analyzing existing operations, those already done. Since I was inexperienced, my ideas differed greatly from what should be, but I learned. Ideals that seem so perfect in college just don't work in the real world. But they are a guide, an invaluable tool."

And Nancy has learned to adapt. Today, still with First County Bank, she is Senior Systems Analyst. When asked what she does now, she replied, "I talk with others, people in banking and business and management to see what we can do to improve our present setup, to work out our problems and plan solutions. If my promotion does happen I'll move into a managerial position where hopefully I can take a more active part in planning, organizing and controlling. I hope to be able to assign personnel to projects and to help others to develop their best potential."

What makes Nancy so enthusiastic about her job? "To me it's not just a 9:00 to 5:00 job bringing home $16,000 a year, it's a career. It's what I want to do..."

Data Processing — if you're looking for a career, try it. It's like the television commercials say, "If it's right you know it, if it's good you feel it..." If Data Processing is for you, you'll know it; you'll want to learn all there is to know and you'll want to apply what you've learned. You may have to work harder than others, but you'll be excited about what you've learned. There's no better feeling than doing something you like to do. ■

SCREENSPLITTER

A COMPLETE TV TEXT DISPLAY SYSTEM from Micro Diversions, Inc.

40 LINES 86 CHARACTERS/LINE

ONBOARD WINDOW SOFTWARE FOR CONTROLLING UP TO 3440 LOGICALLY INDEPENDENT WINDOWS

GENERAL

SCREENSPLITTER is a self-contained hardware/software TV text display system for your microcomputer. It comes on a single, high-quality S-100 buss compatible board, complete with its unique Window Package software module. With the Window Package, you can logically segment SCREENSPLITTER's huge 40 x 86 display of upper-lower case characters into up to 3440 independent "windows" of various sizes. (You get the idea from our ad!) Each window has its own optional frame, cursor, figure-ground, and optional label, and each window scrolls and automatically formats its text independently of all the others. QUICK! There's some interesting information flashing by in WINDOW 1. Go read it!

The Window Package's auto-formatter doesn't care how skinny your windows are (it'll hack your text down to one column if you can stand it!)

WINDOW 1

▪Σ∫÷∂~∈≤↓▲¶ Whoops! Our output routine seems to be having problems. Oh well, at least you get to see some of SCREENSPLITTER'S scientific symbols. (You can order a graphics character set optionally.)

▲▲▲▲▲▲ → time

And any character may be user-defined as a winking character. How? you ask. Simple: SCREENSPLITTER uses a 2708 reprogrammable memory as its character generator. Turn on the character's "wink" bit in the 2708, and presto!

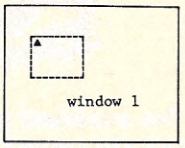
Oh, and naturally, each of the 3440 characters on the screen may have its figure-ground reversed independently.

Frills, you say? No, thrills! Just take a look in the window up there ↗ to see how SCREENSPLITTER puts these raw materials to work in the onboard 1K Window Package▲ (that back there → is the cursor character).

THE CARE AND FEEDING OF WINDOWS

OK. You have just powered on. Initialize the Window Package and turn on your first window:

```
INIT()
OPEN(1,10,15,20,30)
```



Now, just to flex your bits, give the user a wake-up flash (a brief figure-ground reversal inside the window):

```
FLASH(1)
```

Now that you have his attention, go ahead and frame the window (you don't have to, of course):

```
FRAME(1)
```

and, while you're at it, label it, and set the scroll line count:

```
LABEL(1,"General I/O")
SCROLL(1,5)
```

(i.e., when the window fills up, pop it up 5 blank lines)

Just to keep him interested, switch the cursor character from the default caret to the winking caret:

```
CURSORCHAR(1,▲)
```

Now that he's all excited, eyes bulging from the initial flash, transfixed by the hypnotic winking cursor, hit him with some text through window 1:

```
PRINT(1,"I hate to tell you this, William, but
        last night the kids wired that chair
        you're sitting in with 110 volts AC.")
```

Now (this'll really kill him), open a second window to the right:

```
OPEN(2,10,50,5,20)
FRAME(2)
LABEL(2,"Will's Will")
```

and print out a second message through this new window:

```
PRINT(2,"Please type your last will and testament.")
```

Now, of course, you echo his input through window 2, relying on the default scrolling of 1-line "pop-up" when the window fills up.

And on, and on...

PARTIAL FUNCTION SUMMARY

```
INIT()
OPEN(W,X,DX,Y,DY)
CLEAR(W)
FRAME(W,C1,C2,C3)
UNFRAME(W)
REFRAME(W,C1,C2,C3)
LABEL(W,STR,LEN)
LABELS(W,STR)
FLASH(W)
COMPLEMENT(W)
SCROLL(W,N)
CURSORCHAR(W,C)
PRINT(W,STR,LEN)
PRINTS(W,STR)
BACKSPACE(W)
CLEARLINE(W)
FRESHLINE(W)
PLOT(W,X,Y,C)
MOVEWINDOW(W,X,Y,C)
```

POINTS OF INTEREST

- Entire hardware/software system on a single, high-quality S-100 buss compatible board.
- Drives a 10mhz or better TV monitor via standard 75 ohm coaxial cable (supplied).
- 4K static RAM -2114's- display buffer is memory-mapped into your CPU's address space for fast, convenient access if you ever need to bypass the Window Package software.
- User-selectable wait state for operation with 4mhz CPU's.
- 1K onboard 2708 is jumper changeable to a 2K 2716 for user extensions to the Window Package.
- Board presents one TTL load to host, yet drives up to 20 TTL loads via 74367 buffers.
- Provisions for jumpering TV data, sync, blanking off board for external mixing (via 16 pin socket).

SOME APPLICATIONS

1. You have a BASIC program. Open a number of windows, giving each important subroutine in the program its own window. When your program runs, you get a two-dimensional feel of the flow of the execution - flurries of activity here, brief flashes there. You can have the feeling of being able to converse with each subroutine individually!
2. You have a page-oriented text editor. Pick up a paragraph here, a paragraph there, isolating each in its own window while you rummage through the main text in its own large window. Using the MOVEWINDOW function, you can move blocks of text around to produce a final layout.
3. You have an assembly language debugger. Allocate one window to the real-time clock, another to the run-time clock, and several more to display various registers in your 8080 or Z80. Then, you can keep the debugging information separate from your program's I/O, with the debugging information continually present.
4. You have some fancy games. Give each player his own window and define some "community windows." Let your imagination take over!

PS: Watch for our full graphics board, also with its own onboard software... Coming soon!

WHAT YOU GET

- Complete SCREENSPLITTER Kit, with all IC's, low profile sockets, preprogrammed Window Package EPROM, assembly instructions
- Comprehensive Theory of Operation Manual
- Complete source-code listing, and User's Manual for the Window Package
- 90 day warranty on parts and labor

```
+
+
+ + +
+ + +
+ + + + +
```

the PLOT(W,X,Y,"+") thickens

ORDERING INFORMATION

1. Tell us for which 8K boundary you would like your Window Package assembled.
2. Tell us whether you want the scientific symbols, or the graphics characters in ASCII codes 0-31 of your character generator, or the optional APL character generator.
3. Send us a personal check, Master Charge or BAC/VISA number and expiration date. Kit price is $329. Assembled, $429. (Virginia residents please add sales tax.)
4. We will send you the SCREENSPLITTER, postpaid in the continental U.S., from stock to 40 days.

Micro Diversions, Inc.

7900 Westpark Drive, Suite 308
Mc Lean, Virginia 22101
(703) 827-0888

DEALER INQUIRIES INVITED;
EVALUATION KITS AVAILABLE

TIME 06:01:48

What the Computer Taught Me About My Students...

Anne Pasquino

The increased availability of digital computers in the classroom presents a challenge to mathematics teachers at all levels. Generally this challenge concerns how to make the most effective use of the computer in the existing syllabus. Answers to this challenge range from using the computer as a super desk-calculator to developing sophisticated materials such as those produced by the students and teachers involved in Project SOLO.

In the rush to do "impressive things" with the computer and thereby assure school administrators that their dollars have been well spent, we run the risk of overlooking an important and subtle byproduct of computers in the classroom. The student-written program provides us with the opportunity to scrutinize the thought processes of our students and gain some valuable insight into the way they attack problems. In attempting to instruct the computer to perform calculations necessary for the solution of a problem, the student's program mirrors his own problem solving technique or lack of it.

The following problem was assigned to a class of college freshman, non-mathematics majors and produced some interesting results.

> Write a computer program to play the game "I am thinking of a number." The user picks an integer between 1 and 10,000 inclusive. The computer tries to guess the number the user has in mind. The user responds to each guess by indicating whether the guess is too high (type in a 1), low (type in a -1) or correct (type in a 0). The game continues until the computer guesses correctly.

This problem is often given to beginning programming students and can be found in *Getting Started in Classroom Computing,* published by Digital Equipment Corp. As every smug and self-satisfied math teacher knows, the "natural" way to solve this problem is with a binary search. A binary-search procedure assumes a set of items ordered in some logical sequence; in this case, a numerically increasing sequence of numbers. The number sought is compared to the *midpoint* of the set; unless this is the number sought, this number will be found in either the right-hand or the left-hand half of the set. The number sought is compared to the midpoint of the correct half; if not equal to the midpoint, it is then in either the right or the left-hand half of that portion of the set, that is, in one of the two quarters of the set. This procedure is carried out until the number is found.[1]

[1]Philip B. Jordain, *Condensed Computer Encyclopedia,* (New York, 1969), p. 57.

Anne Pasquino, Mathematics Dept., State College at Westfield, Westfield, MA 01085.

True to expectations, several students did use this approach. They "taught" the computer to guess systematically, by first selecting an upper and lower bound. The upper (or lower) bound was revised each time the guess was too high (or too low). In this way the student enabled the computer to squeeze down on the correct value. Each guess was computed by averaging the upper and lower bounds. A typical student program is shown below.

```
 10 PRINT "THE NAME OF THE GAME IS:"
 15 PRINT "PICK A NUMBER FROM 1 TO 10000."
 20 PRINT "THE RULES ARE AS FOLLOWS:"
 30 PRINT "IF MY GUESS IS LOW, TYPE -1."
 40 PRINT "IF MY GUESS IS HIGH, TYPE 1."
 50 PRINT "IF MY GUESS IS CORRECT, TYPE 0."
 60 PRINT
 70 PRINT "PICK YOUR NUMBER"
 80 LET U = 10000
 90 LET L = 0
100 LET G = INT((U + L)/2)
110 PRINT "IS THE NUMBER"; G
120 INPUT A
130 IF A = 0 THEN 190
140 IF A = 1 THEN 170
150 LET L = G
160 GO TO 100
170 LET U = G
180 GO TO 100
190 PRINT "I GUESSED IT."
999 END
```

Not so true to expectations were a number of students who used a dichotomous search, but not a binary search. For example, one student used a first guess of 10,000 and then subtracted 1,000 from the first guess to get the next guess. If the subsequent guess was too high, 1,000 was again subtracted to obtain a new guess. This continued until a response of "too low" was obtained. At this point the last guess was increased in steps of 100 until the guess became "too high." The guess was then lowered in steps of 10 until it became "too low" and finally, increased in steps of one till it was correct. Hence, trapping the correct value was accomplished by adding or subtracting powers of ten to the upper and lower bounds rather than averaging them. The mental decision tree used by the student is pictured in Figure 1.

A similar but more elaborate decision pattern was used by another student who wrote a somewhat longer program; see Figure 2 for the pattern.

Still another student used lower and upper bounds that were adjusted by adding or subtracting an increment. The increment was calculated by a process which resembles the "limiting process" in calculus.

... or Is Binary Search "Natural"?

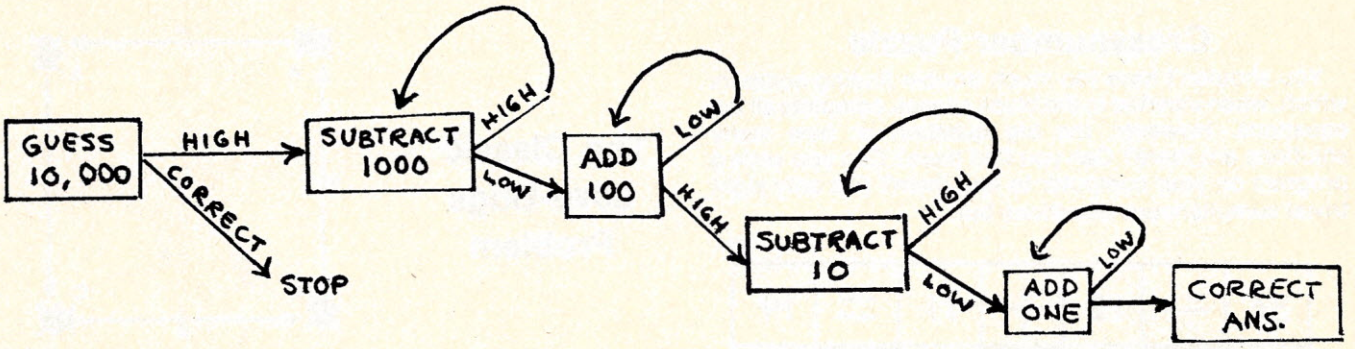


Fig. 1. One student's mental decision tree.

```
10 LET L = 0
15 LET U = 10000
20 LET G = 0
30 FOR N = 1 to 14
35 PRINT "IS THE NUMBER"; G
40 INPUT A
45 IF A = 0 THEN 98
50 LET H = INT(10000/2 ↑ N)
55 IF A = 1 THEN 75
60 LET L = G
65 LET G = L + H
70 GO TO 85
75 LET U = G
80 LET G = U - H
85 NEXT I
98 PRINT "I GUESSED IT."
99 END
```

The value of the increment H in line 50 grows successively smaller with each pass through the loop.

The techniques used by the students in solving this problem were intriquing for two reasons. First, the "natural" application of a binary-search procedure where one continually guesses the midpoint, turned out to be not so "natural." Second, although the students involved lacked formal training in calculus, they seemed to possess an intuitive understanding of the notions of upper and lower bound, convergence and limit, and were able to use complex decision trees. These observations suggest the desirability of inventing a series of "problems" such as the "number game" which might be used to introduce concepts in calculus such as limit, convergence, etc. The problems might also serve as a diagnostic tool to help assess where a student stands regarding such concepts. Further, the use of such problems may reveal that many problem-solving techniques which teachers think are "natural" to student thought processes are learned techniques which are alien to or only remotely related to the student's manner of thinking. At any rate, student-written programs are indicative of a great deal more than just the student's ability to program. ■

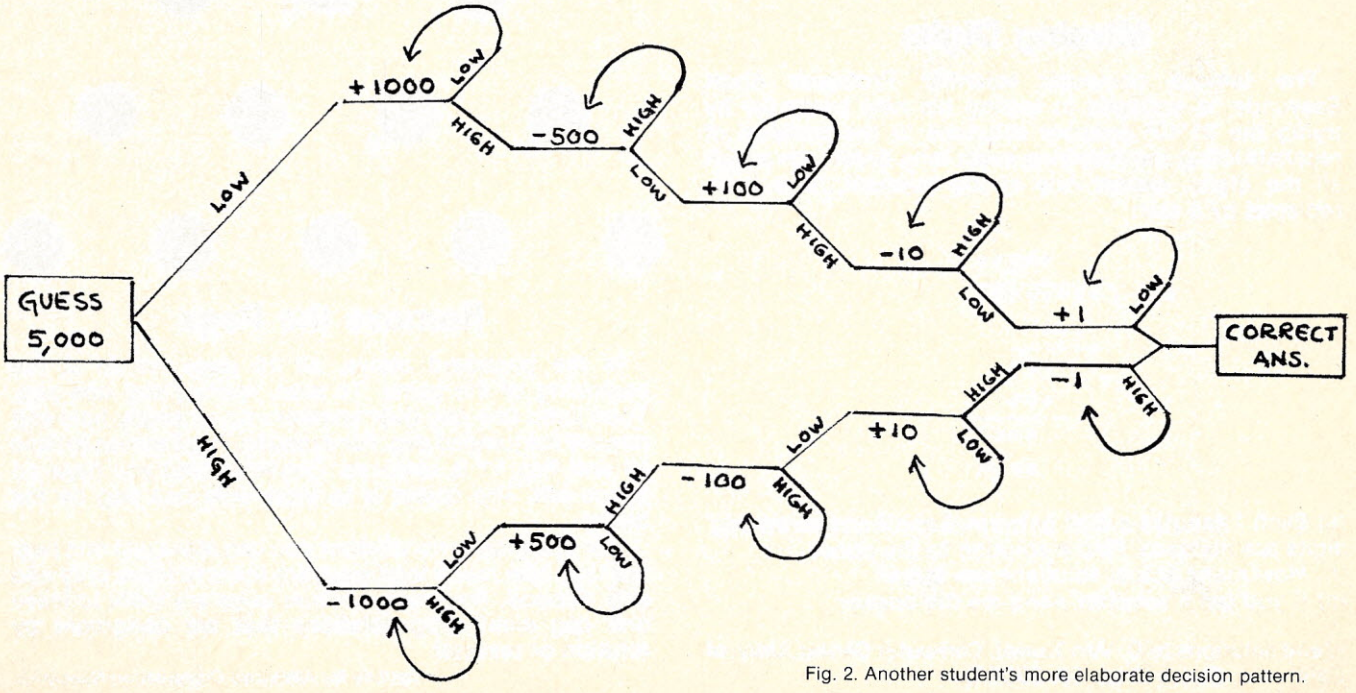


Fig. 2. Another student's more elaborate decision pattern.

puzzles & problems

Crossnumber Puzzle

You shouldn't have too much trouble finding numbers which, when inserted in the blank spaces, complete all the equations. However, the big question is: how many solutions are there? Can you find them? Can you write a program on your computer to find them? (It's only eight trivial sumultaneous equations but...)

DHA

36	÷	12	+	11	=	14
−	■	+	■	−	■	+
	−		×		=	
÷	■	÷	■	−	■	÷
	+		−		=	
=	■	=	■	=	■	=
7	×		÷		=	14

Missing Digits

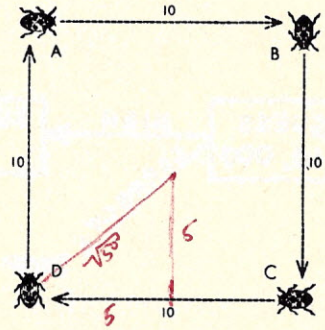
The famous computer scientist, Professor Abort Easycode, is engaged in testing his new computer by trying the $81*10^9$ possible solutions to the problem of reconstructing the following exact long division in which all the digits, except one in the quotient, have been replaced by a star:

```
        **8**
*** / ********
      ***
      ----
       ****
        ***
       ----
        ****
        ****
```

(a) Each * denotes a digit between 0 and 9 and all leading digits are nonzero. Find a solution to the above.
(b) How many actual solutions are there?
(c) If you get a solution, send me the answer.

(Send solutions to D. Van Tassel, Computer Center, Univ. of California, Santa Cruz, CA 95064).

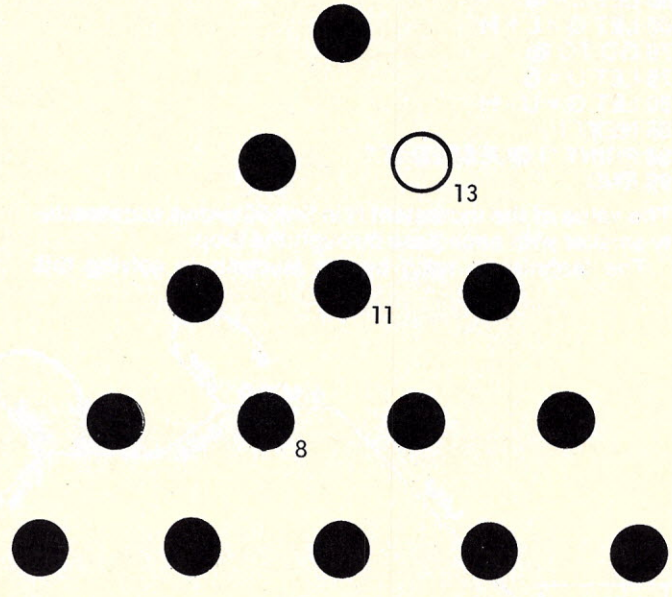
The Classic Love-Bugs Problem



Four bugs, A, B, C, and D, occupy the corners of a square 10 inches on a side. A and C are male, B and D are female. Simultaneously A crawls directly toward B, B toward C, C toward D, and D toward A. If all four bugs crawl at the same constant rate, they will describe four congruent logarithmic spirals which meet at the center of the square.

How far does each bug travel before they meet? (The problem can be solved without calculus.)

Martin Gardner in
Mathematical Puzzles & Diversions



Remove the Pegs

In the pegboard above, all 15 pegs are in at the beginning of the game. To start, remove any one peg. Then jump one peg over another into an empty hole and remove the jumped peg. For example, Peg 8 moves to Hole 13 and Peg 11 is removed. Continue until you have no jumps left. The object is to leave only one peg on the board.

Our question is no whether you can leave just one peg, but first *how many total ways* are there to leave only one peg. Second, *how many unique ways* are there to leave one peg eliminating solutions that are congruent by rotation or reversal.

Institute for Advanced Computation Newsletter

GRAND OPENING: APRIL 28

COMPUTER COMPONENTS OF ORANGE COUNTY

(FORMERLY COMPUTER PLAYGROUND)

- **ORANGE COUNTY'S ONLY AUTHORIZED COMMODORE PET DEALER.**
- PRODUCTS TO BE CARRIED: IMSAI, VECTOR GRAPHICS, APPLE, TDL, POLYMORPHIC, PROCESSOR TECHNOLOGY, CROMENCO, PERSCI, CENTRONICS, TARBELL, VISTA, SHUGART, NORTH STAR, HURISTICS, COMPUTER WORLD, KOYO MONITORS, KIM, MICRO-DESIGN AND MANY MANY MORE . . . . . .
- OUR NEW LOCATION IS NEXT DOOR FROM WHERE WE PRESENTLY ARE.
- WE WILL BE THE **LARGEST** (OVER 5000 SQUARE FEET) AND **MOST COMPLETE** COMPUTER STORE IN SOUTHERN CALIFORNIA.
- BROWSE THROUGH THE **LARGEST MICROCOMPUTER BOOK AND MAGAZINE COLLECTION** ON THE WEST COAST.
- WE ALSO **TEACH CLASSES IN BASIC AND ASSEMBLY LANGUAGE** AS WELL AS **PROVIDE SERVICE AND REPAIR.**

GRAND OPENING DISCOUNTS

- Microcomputer books, Magazines, Software, Peripherals, Components Kits, Systems 10% Off
- Electronic Games 50% Off
- More Discounts on Other Items.
- All Discount Purchases Must Be Prepaid. No Charges.
- Offer Good Thru June 3, 1978.

Computer Components of Orange County
6791 Westminster Ave.
Westminster, Calif. 92708
(714) 898-8330

Other Computer Component Stores Located at:

- 5848 Sepulveda Blvd.
  Van Nuys, CA 91411
  (213) 786-7411
- 4705 Artesia Blvd.
  Lawndale, CA 90260
  (213) 370-4842

CIRCLE 156 ON READER SERVICE CARD

The Perfect Puzzle for Computer Mathematics?

Lynn D. Yarbrough

A puzzle is a problem with a little fun in either its statement or its solution. The fun, for those who are entertained by such things, arises out of some challenge to the imagination that the puzzle presents. A puzzle that requires the use of a computer for its solution is a rarity: most puzzles rely for their fun on the statement of the problem, while most of the rest are interesting because of some ingenuity in the use of analysis or logic in their solution. We are just beginning to see the popularization of puzzles that appeal to the computer scientist because of the complexity or length of the computation required to find the solution.

Such puzzles should require intelligent use of the computer, rather than simple brute-force methods. For example, many combinatorial problems have simple solutions that require inordinate amounts of computer time, but at the same time can be solved very quickly by the use of some combinatorial tricks or shortcuts.

Puzzles are usually characterized by being specific rather than general, so that the solution method may depend for its success on some peculiar property of the values of variables in the problem statement. Thus, while the solution of a differential equation by computer is of great utility, such a problem is of little interest as a puzzle (because of the existence of "canned" routines for the solution) unless there is something unusual about the particular equation to be solved.

Here are my criteria for the ideal puzzle for computer solution:

1. The puzzle should be briefly stated, in terms that can be grasped by, say, a high-school student.
2. The puzzle should not be open to completely analytical solution, since otherwise the need for the computer would be circumvented. At the same time, the solution should be, for all practical purposes, inaccessible by hand calculation. Instead, the puzzle should tax the arithmetic capabilities of the computer at hand, requiring some planning to avoid both excessive running time as well as the traps of roundoff or truncation errors and other anomalies.
3. The correct solution should be readily verifiable; for example, by use of a pocket calculator or other readily accessible device.
4. The investigation of the puzzle should lead the solver naturally into subproblems of independent interest.
5. It wouldn't hurt if the puzzle had some historical or personal interest, beyond its challenge as a problem to be solved.

The following problem satisfies all the above criteria and is my candidate for the ideal computer puzzle; the subsequent problems are also of interest but are deficient in some regard, as indicated.

Problem 1

Find three distinct right triangles with the following properties:

1. The triangles are Pythagorean; that is, all three sides are integers.
2. The perimeters of the three triangles are equal.
3. The areas of the three triangles are in arithmetic progression.

How does this problem stack up against the criteria?

1. Judge for yourself the simplicity of statement.

for a college-level Computer Science class.

2. The minimum solution I know of, which (for those of us who cheat) appears in Dickson's *History of the Theory of Numbers*, Vol. II, consists of nine sides, each of which is an integer eight digits long. The common perimeter is also eight digits long and the three areas are 15 digits long.
3. The solution can be verified by pocket calculator: The calculation of all 15 digits of the area can be circumvented by factoring out the semi-perimeter (half the perimeter) from the area formula, which reduces the calculation to a reasonable size.
4. One solution I have worked out in some detail involves the subproblems of generating all Pythagorean triangles, of factoring large integers, of enumerating the 3-subsets of a set, and of sorting. Some tree-trimming methods, for reducing the number of triangles to be enumerated, are also involved. The puzzle is a study in combinatorial methods.
5. The first solution of this problem was obtained by hand and published in 1819! There is no information available on how long it took to solve in this way. This is not an easy problem; it is perhaps in the nature of a term problem

Problem 2

Find the smallest solution in positive integers *x* and *y* of

x**2 - N*Y**2 = 1, where N = 61.

A. H. Beiler's *Recreations in the Theory of Numbers* contains a straightforward algorithm for the solution of this (Pellian) type of equation, based on the theory of continued fractions. This particular problem (that is, for N = 61) has a solution 10 digits long. (The length of the solution varies unpredictably with N. If you want a real challenge, try the case N = 9781, for which x and y are each 150 digits long!)

This problem is less interesting than problem 1, especially in the number of subproblems involved in the solution. It is practicable, using a pocket calculator and knowing the algorithm, to solve by hand in a reasonably short time.

Problem 3

In how many ways can the integer 10,000 be expressed as a sum of distinct positive integers (ignoring permutations)?

This problem is a good one for introducing backtrack methods of solution and can be solved on a relatively small computer if the proper representation and methods are used. Verification is very laborious; the only practical way is to verify the algorithm for values less than 10,000 and prove that the increased value does not introduce problems.

Problem 4

Find the minimum value of the Gamma function (Gamma(n) = (n-1)! for integer n) in the range 1< n <2.

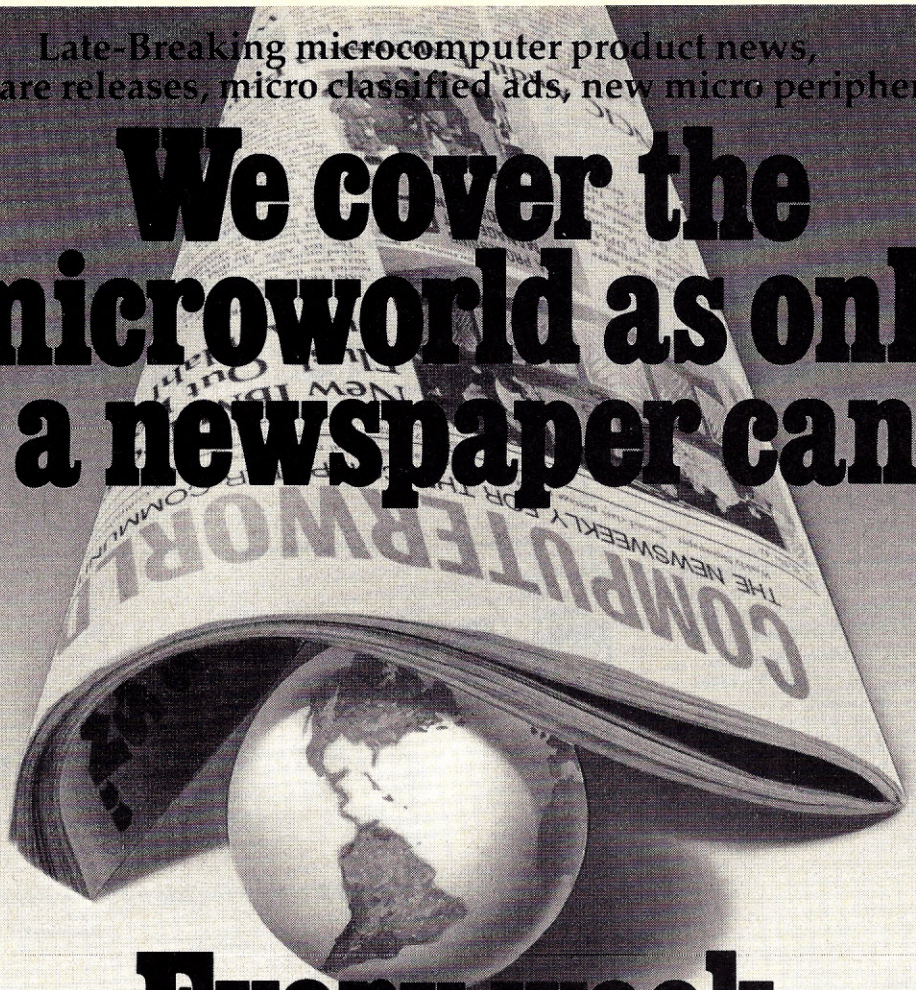
You are not likely to find a subroutine to calculate the Gamma function in your subroutine library (there is a nice algorithm in Henrici's *Computational Analysis on the HP-25 Programmable Calculator)* and you won't be able to use Newton's Method since evaluating the derivative of the Gamma function is another hard problem. Some investigation of the error in evaluating the function will be required to assure a correct solution.

This problem is an interesting one, requiring some resourcefulness, but probably beyond the grasp of most high-school students. ■

Lynn D. Yarbrough, 128 Simons Road, Lexington, MA 02173.

Late-Breaking microcomputer product news,
software releases, micro classified ads, new micro peripherals . . .

We cover the microworld as only a newspaper can—



Every week,

Monthly Magazines are great for in-depth articles and detailed technical information—and we're not suggesting you stop reading this one. What we *are* suggesting is that we can give you a lot that a magazine can't.

First, we're a newspaper—which means we cover a lot of things briefly, instead of a few things deeply.

Second, we're weekly—so you won't be saying "I wish I had known about that" when you hear someone talking about new micro hardware and software.

Third, we have a Microcomputing Classified Exchange, where you can find (or advertise) bargains on equipment or software—or look for people or information you need.

Fourth, we're into all aspects of computers; so you don't have to limit your information to micros. We cover everything from industry news to data processing to ethical issues, as only a newspaper can.

Fifth, we'll send you 52 issues for about 35c a week with a one-year subscription (less than half the newsstand price). Just use the form below to start your subscription coming. It's a fast way to stay ahead.

Please send me **COMPUTERWORLD** *for 1 year* RATES: U.S. - $18 Canada and PUAS - $28 Europe & Middle East-$75 All Other Foreign-$125

☐ Check Enclosed
☐ Am Ex ☐ BA/Visa ☐ MC (MC Only - List four digits above your name)

If charge we must have cardholder's signature: _____ Expiration Date: _____

First Initial		Middle Initial	Surname	
Your Title				
Company Name				
Send to Address				
City		State	Zip Code	

Address shown is ☐ *Business* ☐ *Home.*

☐ Check here if you do not wish to receive promotional mail from *Computerworld.*

COMPUTERWORLD
THE NEWSWEEKLY FOR THE COMPUTER COMMUNITY

CIRCULATION DEPT. CC., 797 Washington Street, Newton, Mass. 02160

PLEASE CIRCLE 1 NUMBER IN EACH CATEGORY

BUSINESS/INDUSTRY
10 Manufacturer of Computer or DP Hardware/ Peripherals
20 Manufacturer (other)
30 DP Service Bureau/Software/Planning Consulting
40 Public Utility/Communication Systems/ Transportation
50 Wholesale/Retail Trade
60 Finance/Insurance/Real Estate
70 Mining/Construction/Petroleum/Refining
75 Business Service (except DP)
80 Education/Medicine/Law
85 Government/Federal/State/Local
90 Printing/Publishing/Other Communication Service
95 Other

TITLE/OCCUPATION/FUNCTION
11 President/Owner/Partner/General Manager
12 VP/Assistant VP
13 Treasurer/Controller/Finance Officer
21 Director Manager of Operation/Planning/ Administrative Service
22 Director/ Manager/Supervisor DP
23 Systems Manager/Systems Analyst
31 Manager/ Supervisor Programming
32 Programmer/Methods Analyst
41 Application Engineer
42 Other Engineering
51 Mfg. Sales Representative
52 Other Sales/Marketing
60 Consultant
70 Lawyer/Accountant
80 Librarian/Educator/Student
90 Other

CIRCLE 113 ON READER SERVICE CARD

Computer Conversations

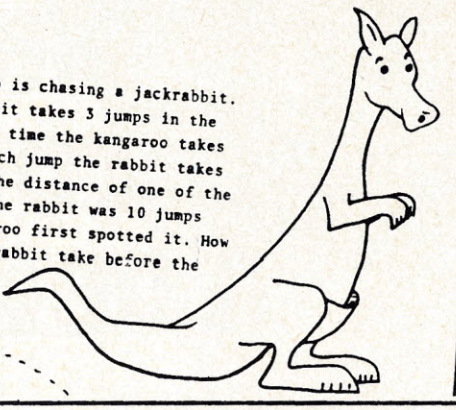
44

$$A+B+C=A*B*C$$

Can you find three natural numbers that give the same result when added or multiplied?

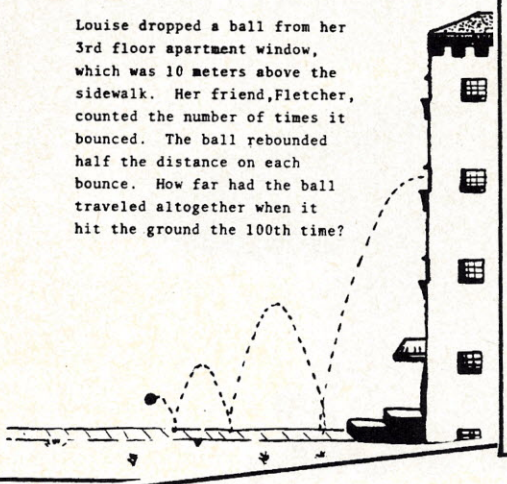
49

A kangaroo is chasing a jackrabbit. The jackrabbit takes 3 jumps in the same length of time the kangaroo takes 2 jumps. But each jump the rabbit takes covers only half the distance of one of the kangaroo's jumps. The rabbit was 10 jumps ahead when the kangaroo first spotted it. How many jumps will the rabbit take before the kangaroo catches it?



52

Louise dropped a ball from her 3rd floor apartment window, which was 10 meters above the sidewalk. Her friend, Fletcher, counted the number of times it bounced. The ball rebounded half the distance on each bounce. How far had the ball traveled altogether when it hit the ground the 100th time?



62

A, B, C are each a single digit.
What is the minimum value of ABC divided by A+B+C ?
(The answer is not 1.)

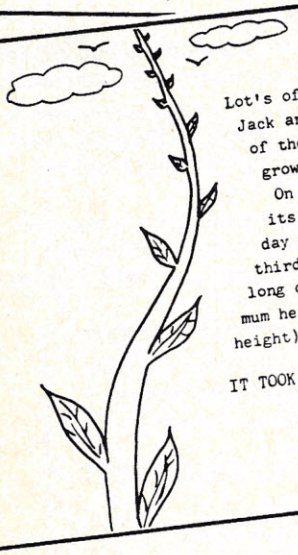
13

K434KO

What value of K would make K434KO divisible by 36?

K = ____

22

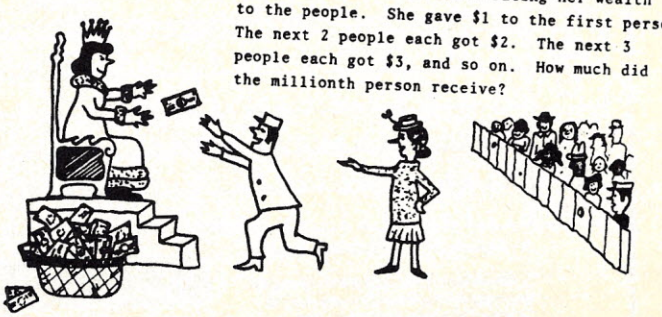


Lot's of people have heard about Jack and his Beanstalk. But most of them don't know about the growth pattern of the beanstalk. On the first day it increased its height by $\frac{1}{2}$. On the second day it increased by $\frac{1}{3}$, on the third day by $\frac{1}{4}$, and so on. How long did it take to reach its maximum height (100 times its original height)?

IT TOOK _____ DAYS

67

A million people lived in the land of BASIC. Queen Terminalla was distributing her wealth to the people. She gave $1 to the first person. The next 2 people each got $2. The next 3 people each got $3, and so on. How much did the millionth person receive?



These activities are reprinted from "Computer Conversations" (a set of 41 colorful 14x21 cm cards) and "More Computer Conversations" (27 cards). "Computer Conversations" costs $3.95, teacher guide $2.95, "More Computer Conversations", $2.95, teacher guide $2.50. Postage on all orders $1.00. The Math Group, 5625 Girard Ave. So., Minneapolis, MN 55419.

ELECTRONIC SYSTEMS

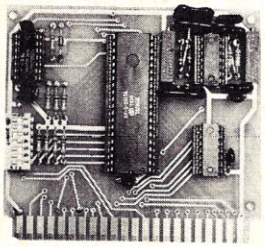
P.O. Box 9641 San Jose, CA 95157 (408) 374-5984

FOR CATALOG INCLUDING PARTS LISTS AND SCHEMATICS, SEND A SELF ADDRESSED ENVELOPE WITH 24¢ POSTAGE.

RS-232/TTY* INTERFACE NEW

Part no. 600

- Converts RS-232 to 20mA current loop, and 20mA current loop to RS-232
- Two separate circuits
- Requires +12 and -12 volts
- Board only $4.50, with parts $7.00

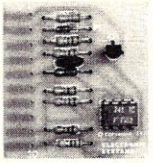


UART & BAUD RATE GENERATOR*

Part no. 101

- Converts serial to parallel and parallel to serial
- Low cost on board baud rate generator
- Baud rates: 110, 150, 300, 600, 1200, and 2400
- Low power drain +5 volts and -12 volts required
- TTL compatible
- All characters contain a start bit, 5 to 8 data bits, 1 or 2 stop bits, and either odd or even parity.
- All connections go to a 44 pin gold plated edge connector
- Board only $12.00; with parts $35.00

RS-232/TTL INTERFACE*



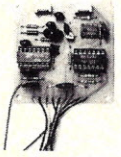
Part no. 232

- Converts TTL to RS-232, and converts RS-232 to TTL
- Two separate circuits
- Requires -12 and +12 volts
- All connections go to a 10 pin gold plated edge connector
- Board only $4.50; with parts $7.00

DC POWER SUPPLY*

Part no. 6085

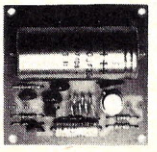
- Board supplies a regulated +5 volts at 3 amps., +12, -12, and -5 volts at 1 amp.
- Power required is 8 volts AC at 3 amps., and 24 volts AC C.T. at 1.5 amps.
- Board only $12.50; with parts $42.50 excluding transformers



Part no. 111

TAPE INTERFACE*

- Play and record Kansas City Standard tapes
- Converts a low cost tape recorder to a digital recorder
- Works up to 1200 baud
- Digital in and out are TTL-serial
- Output of board connects to mic. in of recorder
- Earphone of recorder connects to input on board
- Requires +5 volts, low power drain
- Board $7.60; with parts $27.50
- No coils

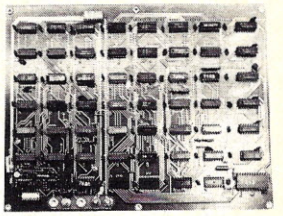


Part no. 107

RF MODULATOR*

- Converts video to AM modulated RF, Channels 2 or 3
- Power required is 12 volts AC C.T., or +5 volts DC
- Board $7.60; with parts $13.50

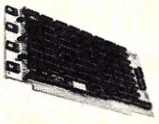
TELEVISION TYPEWRITER



Part no. 106

- Stand alone TVT
- 32 char/line, 16 lines, modifications for 64 char/line included
- Parallel ASCII (TTL) input
- Video output
- 1K on board memory
- Output for computer controlled curser
- Auto scroll
- Non-destructive curser
- Curser inputs: up, down, left, right, home, EOL, EOS
- Scroll up, down
- Requires +5 volts at 1.5 amps, and -12 volts at 30 mA
- All 7400, TTL chips
- Char. gen. 2513
- Upper case only
- Board only $39.00; with parts $145.00

8K STATIC RAM



Part no. 300

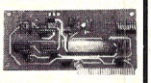
- 8K Altair bus memory
- Uses 2102 Static memory chips
- Memory protect
- Gold contacts
- Wait states
- On board regulator
- S-100 bus compatible
- Vector input option
- TRI state buffered
- Board only $22.50; with parts $160.00

TIDMA*

Part no. 112

- Tape Interface Direct Memory Access
- Record and play programs without bootstrap loader (no prom) has FSK encoder/decoder for direct connections to low cost recorder at 1200 baud rate, and direct connections for inputs and outputs to a digital recorder at any baud rate.
- S-100 bus compatible
- Board only $35.00; with parts $110.00

Apple II Serial I/O Interface*



Part No. 2

- Baud rates up to 30,000
- Plugs into Apple Peripheral connector
- Low-current drain
- RS-232 Input and Output

SOFTWARE

- Input and Output routine from monitor or BASIC to teletype or other serial printer.
- Program for using an Apple II for a video or an intelligent terminal. Board only – $15.00; with parts – $42.00; assembled and tested – $62.00.

MODEM*

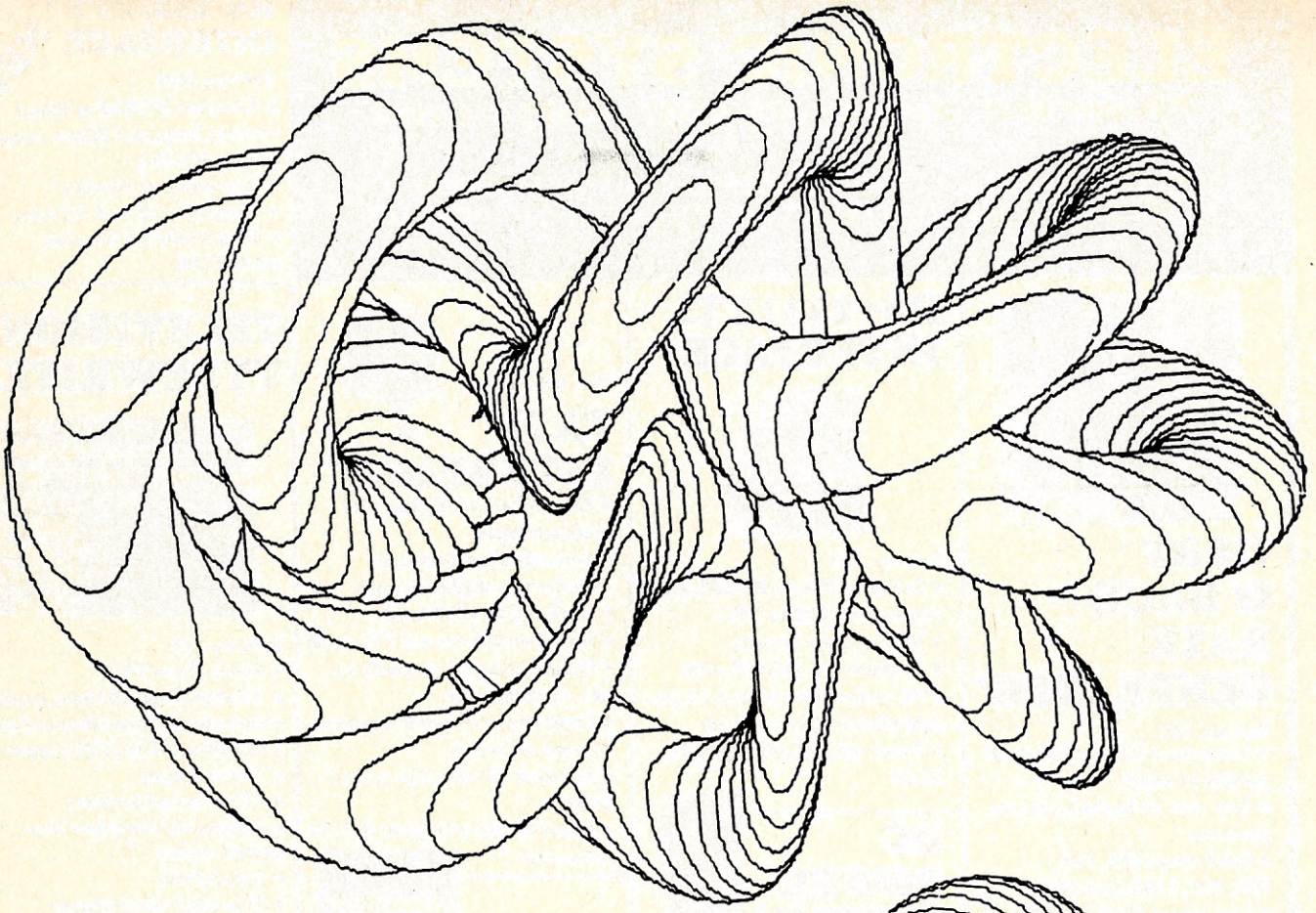
Part no. 109

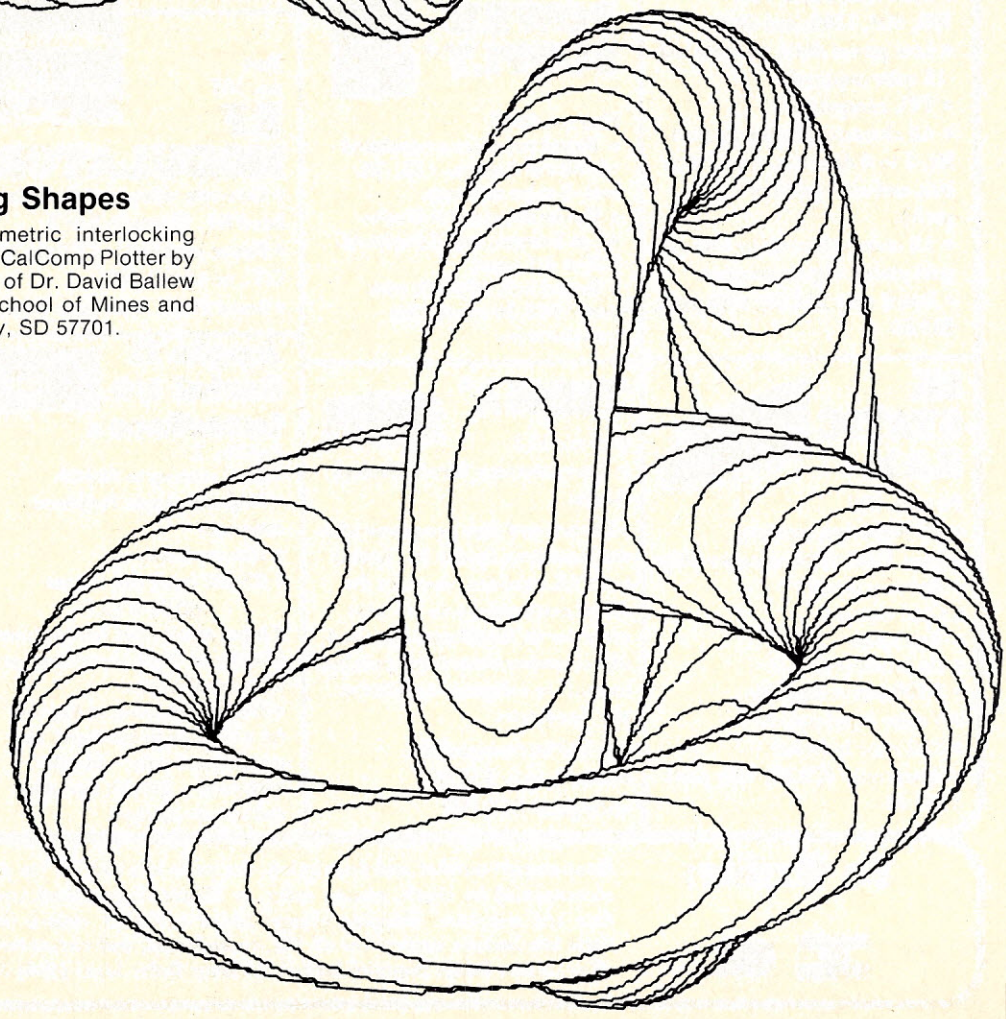
- Type 103
- Full or half duplex
- Works up to 300 baud
- Originate or Answer
- No coils, only low cost components
- TTL input and output-serial
- Connect 8 ohm speaker and crystal mic. directly to board
- Uses XR FSK demodulator
- Requires +5 volts
- Board $7.60; with parts $27.50

To Order:

VISA master charge

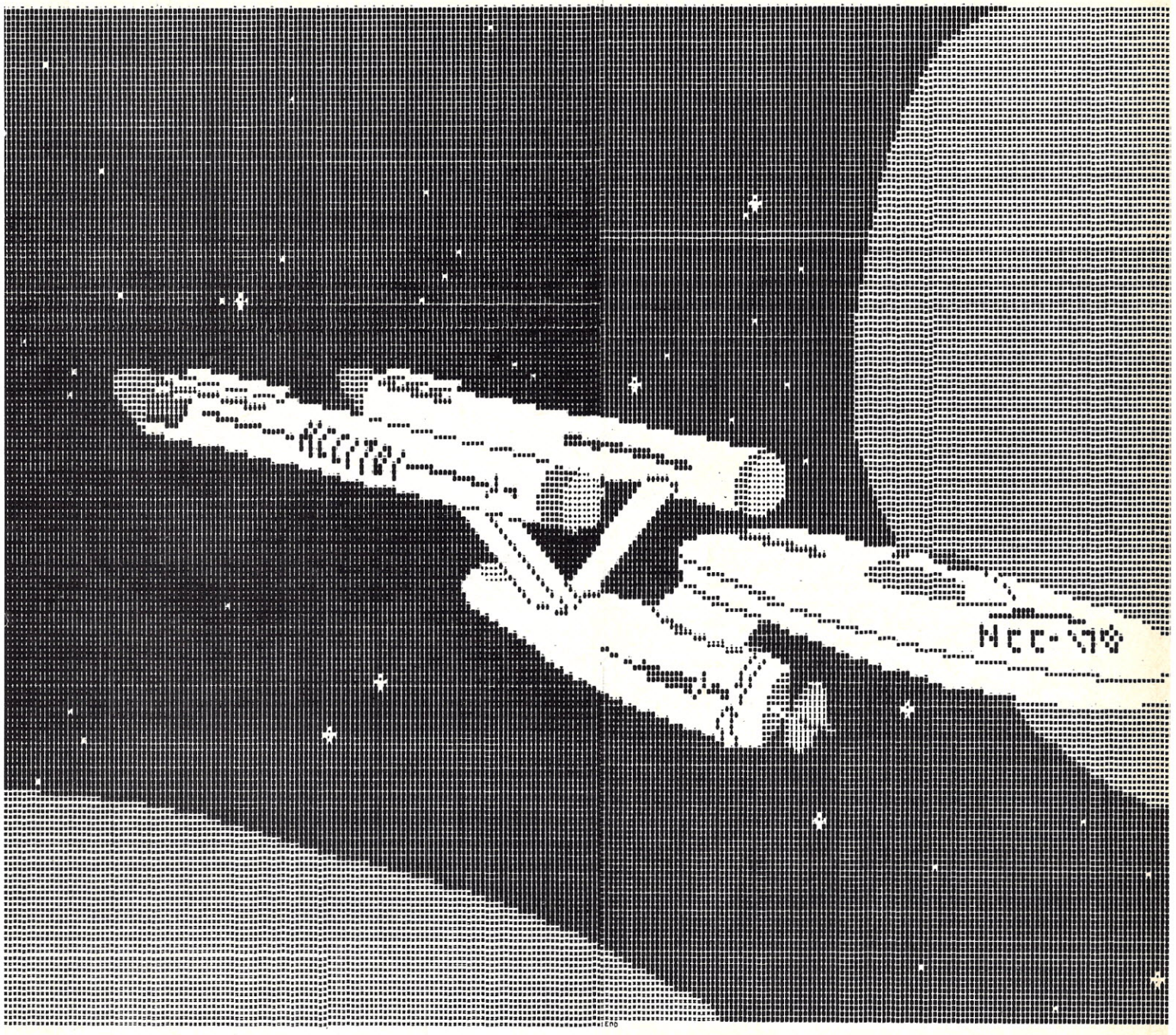
Mention part number and description. For parts kits add "A" to part number. Shipping paid for orders accompanied by check, money order, or Master Charge, BankAmericard, or VISA number, expiration date and signature. Shipping charges added to C.O.D. orders. California residents add 6.5% for tax. Parts kits include sockets for all ICs, components, and circuit board. Documentation is included with all products. Dealer inquiries invited. 24 Hour Order Line: (408) 374-5984.* Designed by John Bell.





Interlocking Shapes

These amazing geometric interlocking shapes were done on a CalComp Plotter by Clark Dailey, a student of Dr. David Ballew at the South Dakota School of Mines and Technology, Rapid City, SD 57701.



Enterprise in Space

This "computer" picture was done by hand by John L. Joseph. He writes, "I am currently attempting to develop an incremental digitizing device, with relatively little success. Could you refer me to a source of such information." If readers can help, write John Joseph, Honeywell Information Systems, 5250 West Century Blvd., Los Angeles, CA 90045.

Tracking Right-Angle Searches

In Mazes and Games

Scott Davidson

This article discusses techniques for searching grid structures, to discover or trace a path. As an example, the WATCHMAN game by Mac Oglesby is taken from a previous issue of Creative. *(In WATCHMAN, you try to walk on each street of a city, once and only once. Obviously, it is permissible to visit the same corner more than once.) The ideas presented here also apply to mazes, word puzzles, and the like.*

Right-angle searches are used to test the values of adjacent horizontal and vertical positions on a grid. In contrast to line-by-line searches, indexing is awkward despite the apparent simplicity. This is evidenced by numerous programs that use separate code for each of the four compass directions. We will start here and then describe some alternatives. The focus will be on situations where we wish to trace a connected horizontal and vertical path on a grid. An example is the WATCHMAN program (*Creative Computing*, Sept-Oct 1976, p 74-75).

If the present position can be on the periphery of the grid, an out-of-bounds check is required for each direction before a search test can be made. This situation can be avoided by expanding both X and Y dimensions by two and framing the original grid with a non-search value. (Sometimes this frame can be used to enhance the grid display.) The routines below assume that this has been done.

I Clockwise—four IF statements

```
5   REM ---- PRESENT POSITION IS M(I, J) T = TEST VALUE
.
.
.
100 IF M(I,J+1)<>T THEN 140 'LOOK RIGHT
110 I2=I
120 J2=J+1
130 GO TO 400
140 IF M(I+1,J)<>T THEN 180 'LOOK DOWN
.
.
.
300 REM *** NO - FIND ACTION
.
.
.
400 I=I2
410 J=J2
420 GO TO 100
```

Sample Run

```
VILLAGE MAP:

      1*********2
      * 1ST ST. *
      *        *
******3*********4******
*1   2* 2ND ST. *3   4*
*S   N*         *R   T*
*T   D* 3RD ST. *D   H*
*     5*********6     *
*A   A*         *A   A*
*V   V*         *V   V*
*E   E* 4TH ST. *E   E*
*N    7*********8******
*U              *
*E      5TH ST. *
*****************

START AT WHAT CORNER(1-- 8 ) ?3
ENTER HEADING=0 FOR ROUTE MAP ( :::=FOOTPRINTS)
HEADINGS:  N=1, S=2, E=3, W=4  (MAP=0)
YOU'RE AT CORNER # 3 HEADING ?4
YOU'RE AT CORNER # 8 HEADING ?4
YOU'RE AT CORNER # 7 HEADING ?4
***ILLEGAL MOVE--TRY AGAIN
 ?1
YOU'RE AT CORNER # 5 HEADING ?1
YOU'RE AT CORNER # 3 HEADING ?0

      1*********2
      * 1ST ST. *
      *        *
::::::3*********4******
:1   2: 2ND ST. *3   4*
:S   N:         *R   T*
:T   D: 3RD ST. *D   H*
:     5*********6     *
:A   A:         *A   A*
:V   V:         *V   V*
:E   E: 4TH ST. *E   E*
:N    7:::::::::8******
:U              :
:E      5TH ST. :
::::::::::::::::::

HEADINGS:  N=1, S=2, E=3, W=4  (MAP=0)
YOU'RE AT CORNER # 3 HEADING ?3
YOU'RE AT CORNER # 4 HEADING ?2
YOU'RE AT CORNER # 6 HEADING ?4
***YOU'RE TRAPPED AT CORNER # 5 --WANT FINAL
                               MAP(Y OR N) ?Y
      1*********2
      * 1ST ST. *
      *        *
::::::3:::::::::4******
:1   2: 2ND ST. :3   4*
:S   N:         :R   T*
:T   D: 3RD ST. :D   H*
:     5:::::::::6     *
:A   A:         *A   A*
:V   V:         *V   V*
:E   E: 4TH ST. *E   E*
:N    7:::::::::8******
:U              :
:E      5TH ST. :
::::::::::::::::::

YOU WERE SUPPOSED TO PATROL THE WHOLE VILLAGE!
```

Scott Davidson, 1452 South 3 St., Louisville, KY 40208

The no-find action depends on the application. In tracing a maze, this can be a backtrack move to the previous (stored) position. Generally a tracing search terminates when either a specific goal is reached or all forward moves are blocked.

This code averages 2.5 tests for each find, which is unnecessarily slow if the path being traced has many forward moves for each turn. A better strategy is to look next in the last successful direction. Even if the path is random, this direction is as good as any other and has already been computed. One way to do this is to use the search routine only to find an initial direction, then follow with a short, fast move-until-blocked tracking loop (Ia):

```
100 IF M(I,J+1)<>T THEN 140
110 V1=0
120 H1=1
130 GO TO 400
.
.
.
400 I=I+V1
410 J=J+H1
420 IF M (I+V1,J+H1) =T THEN 400
430 GO TO 100
```

Another way is to do the tracking inside the search routine (Ib). This one "corners" faster than (Ia), but the find action is restricted to moves.

```
70  LET D4=1 'INITIAL DIRECTION - SET ONLY ONCE
80  FOR N=1 TO 4
90  ON D4 THEN 100, 140, 180, 220
100 IF M(I,J+1)<>T THEN 135
110 LET J=J+1
120 GO TO 80
135 LET D4=2
137 GO TO 260
.
.
.
250 LET D4=1
260 NEXT N
```

As a further refinement we note that a backward move yields no net progress and is usually illegal while forward moves or turns are possible. In a clockwise search the third look is backward while the last is a turn. The following fix reverses this (Ic):

```
100 IF M(I,J+1)<>T THEN 255
.
.
.
255 LET D4=D4+N 'ADD N NOT 1
256 IF D4<5 THEN 260
257 LET D4=D4-4
260 NEXT N
```

The four IFs can be reduced to one by indexing, at the sacrifice of speed. The scan increments are treated as irregular using DATA arrays. (Kernighan and Plauger (1974) do this in their PL/I mouse-in-a-maze program).

```
20 MAT READ V(4),H(4)
30  DATA 0,1, 0, -1, 1, 0, -1, 0
.
.
.
100 IF M(I+V(D4),J+H(D4))<>T THEN 255
```

A second approach to the right-angle search is to do it row-by-row. As in the unmodified clockwise search (I), the scan order is fixed, hence this is more suitable for finding an initial direction. In contrast, however, the code is compact and indexes smoothly without DATA arrays:

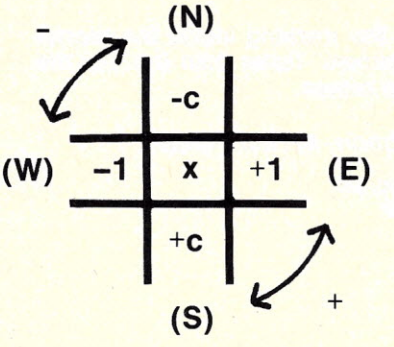
```
00010 REM=====SIMPLIFIED WATCHMAN PROGRAM
00020 DIM A(425),D(23)              'A(17,25)=15X23 FRAMED WITH ZEROES
00030 REM=====GOSUB TO PRINT INSTRUCTIONS (OMITTED)
00040 LET V=ASC(*)                  'VALUE SOUGHT TO STAY ON ROAD
00050 GOSUB 1000                    'READ MAP, COUNT STEPS, LABEL CORNERS
00060 PRINT "VILLAGE MAP:"
00070 GOSUB 2000                    'PRINT MAP
00080 MAT READ B(4)                 'N, S, E, W BEARINGS
00090 DATA -25,25,1,-1
00100 LET C1=26                     'COLUMNS+1
00110 LET D4=1                      'INITIAL SEARCH DIRECTION
00120 PRINT "START AT WHAT CORNER(1--"N")";
00130 INPUT J
00140 LET L=C(J)
00150 PRINT "ENTER HEADING=0 FOR ROUTE MAP ( 1111=FOOTPRINTS)"
00160 PRINT "HEADINGS:  N=1, S=2, E=3, W=4   (MAP=0)"
00170 GOSUB 500                     'SEARCH FOR UNTROD PATH
00180 IF T=0 THEN 390               'NONE FOUND?
00190 PRINT "YOU'RE AT CORNER #"A(L)-48"HEADING";
00200 INPUT H
00210 IF H>0 THEN 240
00220 GOSUB 2000
00230 GO TO 160
00240 LET D4=B(H)
00250 LET C1=26*SGN(D4)
00260 LET H1=D4                     'SAVE INITIAL HEADING
00270 GOSUB 500
00280 IF D4=H1 THEN 330             'NO FORCED TURN?
00290 PRINT "***ILLEGAL MOVE--TRY AGAIN"
00300 GO TO 200
00310 GOSUB 500                     '=====>NEXT STEP
00320 IF T=0 THEN 370               'BLOCKED?
00330 LET L=T                       'MOVE FWD OR TURN
00340 LET A(L)=ASC(!)               'MAKE FOOTPRINT, BLOCK RETRACE STEP
00350 LET S=S+1                     'COUNT STEP
00360 GO TO 310                     '=====>
00370 LET L=L+D4                    'STEP TO CORNER
00380 GO TO 170
00390 PRINT "***YOU'RE TRAPPED AT CORNER #"A(L)-48"--WANT FINAL MAP
00400 INPUT A$                                          (Y OR N)";
00410 IF A$<>"Y" THEN 430
00420 GOSUB 2000
00430 IF S=S9 THEN 460              'ALL STEPS TROD?
00440 PRINT "YOU WERE SUPPOSED TO PATROL THE WHOLE VILLAGE!"
00450 GO TO 9999
00460 PRINT "CONGRATULATIONS!"  'WINNER
00470 GO TO 9999
00500 REM ***SUB*** TRACKING RT-ANGLE SEARCH ON LINEAR ARRAY
00510 FOR I=1 TO 2
00520 FOR J=1 TO 2
00530 LET T=L+D4
00540 IF A(T)<>V THEN 560
00550 RETURN
00560 LET D4=C1-D4
00570 NEXT J
00580 LET D4=D4-C1
00590 LET C1=-C1
00600 NEXT I
00610 LET T=0
00620 RETURN
01000 REM =SUB= INPUT STREET PLAN
01010 LET V0=ASC(0)
01020 FOR I=27 TO 377 STEP 25
01030 READ D$
01040 CHANGE D$ TO D
01050 LET J=0
01060 FOR L=I TO I+22
01070 LET J=J+1
01080 LET T=D(J)
01090 LET A(L)=T
01100 IF T<>V THEN 1120
01110 LET S9=S9+1
01120 IF T<>V0 THEN 1160
01130 LET N=N+1
01140 LET C(N)=L
01150 LET A(L)=N+48
01160 NEXT L
01170 NEXT I
01180 DATA "      0*********0      "
01190 DATA "      * 1ST ST. *      "
01200 DATA "      *         *      "
01210 DATA "******0*********0******"
01220 DATA "*1   2* 2ND ST. *3   4*"
01230 DATA "*S   N*         *R   I*"
01240 DATA "*T   D* 3RD ST. *D   H*"
01250 DATA "*     0*********0     *"
01260 DATA "*A    A*         *A    A*"
01270 DATA "*V    V*         *V    V*"
01280 DATA "*E    E* 4TH ST. *E    E*"
01290 DATA "*N     0*********0******"
01300 DATA "*U       *            "
01310 DATA "*E       5TH ST. *      "
01320 DATA "*****************      "
01330 RETURN
02000 REM =SUB= PRINT STREET MAP
02010 PRINT
02020 FOR I=27 TO 377 STEP 25
02030 FOR J=I TO I+22
02040 PRINT CHR$(A(J));
02050 NEXT J
02060 PRINT
02070 NEXT I
02080 PRINT
02090 RETURN
09999 END
```

Program Listing

```
90  LET Z1=0
100 FOR R=I-1 TO I+1
110 FOR C=J-Z1 TO J+Z1 STEP Z1+1
120 IF M(R,C)= T THEN 400
130 NEXT C
140 LET Z1=1-Z1
150 NEXT R
.
.
.
400 LET I=R 'OR V1=R-I ETC. (TRACKING LOOP)
410 LET J=C
420 GO TO 90
```

Linear Array

A standard programming technique that offers several advantages here is to use a linear array to represent the grid. Any of the above routines will execute faster if this conversion is made. Each move is specified by one instead of two values. In BASIC, interconversion of string and numeric data is simplified. The approach also leads to a third scanning sequence in which the desired ahead—turn—turn order is inherent. The look backward is always last, and if the search fails, the initial search direction is automatically restored. The four directions are grouped as two complementary pairs of complementary right-angle directions if you'll excuse the jargon. Anyway, here is the picture:



To illustrate, the WATCHMAN program has been recoded to use the tracking right-angle search on a linear array as a subroutine. This approach leads to a program which is not only much shorter but also essentially independent of the street plan. For example, the elegant flag array of prime products—used to determine whether all streets have been patrolled—can be replaced by a simple count comparison of watchman steps with total street steps. This will work for any street plan. Similarly, the street corners are located and labeled 1--N during read-in. (Programming problem: write a subroutine that will build a corner location table (ST,AVE) during read-in of any street plan. Note that, in general, column (AVE) numbers must be reassigned after read-in.)

At each new corner we test for an untrod path. If the watchman is trapped, the patrol is finished. (Purists note: this code even works for "Null Village," which has street corners but no streets!) The user need only specify an initial heading, since the next corner will be found automatically regardless of the number of turns in-between. (An illegal move is flagged if the first GOSUB returns a new heading). The footprints made by the watchman are more than decorative; they serve to prevent illegal retracing steps. When the next step is to a corner, the search fails, but since the initial search direction is restored, we simply move the watchman one more step ahead and the loop is closed.

Reference: Kernighan and Plauger, "The Elements of Programming Style," McGraw Hill, NY (1974), p 50-51. ■

3rd Annual MACC Computerfest™ '78 June 23–25, 1978

Presented by the Midwest Affiliation of Computer Clubs

Featuring

- ★ Stupendous Hobbyist Exhibits
- ★ Tours and Evening Activities
- ★ Club Hospitality Suites
- ★ Special Club Meetings
- ★ Fabulous Programs
- ★ Manufactures Party
- ★ Technicial Sessions
- ★ Giant Flea Market
- ★ Free Seminars
- ★ New Products
- ★ Exhibits

Forget everthing you thought you knew about a computerfest. Because this is an all new computerfest in an all new city.

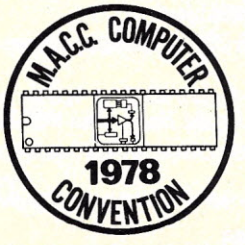
Quite simply, Detroit has what it takes to be the number 1 convention city in America and the MACC is holding the 1978 Computerfest™ in the Renaissance Center–a $500 million total environment complex one-third larger than Rockefeller Center. On 33 acres of landscaped riverfront. Come stay and relax in our new Detroit Plaza, which offers a convention environment unlike anything else on earth. With 70 stories of circular glass walls rising 800 feet, 1404 world class guest rooms, and special rates for our Computerfest™ guests. Secure, covered parking for thousands of cars. There is so much *action* happening during the three days of the fest, you'll want to *bring your walking shoes* when covering the over 100,000 square feet of convention facilities. We have provided for your complete comfort while at the Computerfest™ '78 with the following fantastic facilities *indoors* at the Renaissance Center:

- Eleven restaurants and lounges
- World's largest rooftop trilevel revolving lounge and restaurant complex
- Incredible walkways
- Outstanding recreation facilities
- Fantastic shops

There's even an eight story atrium with cocktail pods, hanging gardens and a giant reflecting pool where you can relax after those tough night programming sessions! Detroit's close to everyone with over 60% of America's population just one air hour away. The Detroit Plaza's located on the banks of the Detroit River. Seconds from Detroit's exotic Windsor tunnel. Within walking distance of adjacent Cobo Hall, Ford Theater, and the Civic Center.

Foreign visitors are invited to fly into our ultra efficient International Metropolitan airport, just 30 minutes from our Computerfest™. For special motel/hotel rates and show reservation information, write: MACC Computerfest™, Box 9578 - Department LIT, Detroit, Michigan 48202 U.S.A. or call our 24 hour Computerfest hotline number (313)775-5320.

Make your *reservations NOW!*



CIRCLE 116 ON READER SERVICE CARD

. reviews . .

The Home Computer Revolution. Ted Nelson. The Distributors, 702 South Michigan, South Bend, IN 46618. 224 pages, paperback. $2. 1977.

Reviewing a book like *The Home Computer Revolution* is not easy. As many readers of *Creative* know, Ted Nelson is the author of the now-classic *Computer Lib/Dream Machines,* so it would be very convenient to say, "Well, Ted Nelson has done it again." In some ways he has, but in other ways *The Home Computer Revolution* has serious flaws. Since *The Home Computer Revolution* was written in a highly subjective style, this will be a highly subjective review.

In some respects, *The Home Computer Revolution* is like *Computer Lib.* It contains zillions of interesting facts, trivia, and anecdotes about personal computing and computing in general. (The title is slightly misleading, since the book is actually about the state of the art in people-oriented computing, of which the home computer is a major part.) So, the book itself should be quite interesting to anyone involved with, or just becoming acquainted with, personal computing. However, one has a sneaking suspicion that a total novice might be dreadfully confused after reading *The Home Computer Revolution.* In one portion of the book, Ted Nelson drags out a whole series of his cutesy terms relating to an idealized interactive graphics screen: menus, menuplexes, panels or windows, areas, prompts, prompt areas, command lines, ding-dongs, ding-dong cursors, pop-ins, peekaboos, and finally, doorbells. And he has the *NERVE* to complain about JCL! Mumbo-jumbo is mumbo-jumbo, whether it is fanciful or serious. And based upon my personal experience with JCL, it is indeed much too difficult to learn, but it is also a powerful tool in the hands of those who understand it. So, *The Home Computer Revolution* is like *Computer Lib* in that it is an interesting collection of interesting facts and stories, told from the Ted Nelson point of view, but brought up to date.

However in some ways *HCR* is not like *CL/DR* at all. Instead of being a nice, freaky, alternative-press type of publication, *The Home Computer Revolution* is a slick paperback with catchy buy-me phrases on the front and back covers. (Cut to a scene in the local Shop-Rite. Child, to mother: "Mommy, look at what just fell out from behind the cling peaches!" Cut back to review.)

Unfortunately, the author of this book has taken a bit too much liberty in describing himself. Example: There is a subheading in the book entitled, "The Far Future (Beyond Five Years)." Under this, in parenthesis, Ted writes, "Anyone who tries to predict beyond five years is crazy." As we all know, it is fashionable to be called crazy, because that really means you're smart and an extraordinary person. REAL crazy people don't act crazy any more; they're locked up and doped up in institutions. Example: On page 44, we read, "A personal note. In my own speeches from 1965 on, I have rarely failed to point out that the real and true market for computers was going to be in the home. People were skeptical." Well, Ted was right. So? It's OK if *other* people say you're a genius, or farsighted, or crazy; but as soon as you start doing it yourself, it doesn't work.

What really drives this reviewer ba-na-nas is the way Ted Nelson goes after IBM. IBM is accused, in almost as many words, of preventing the advent of personal interactive computing by the introduction of the System 360/370. Nelson says that IBM computers are all wrong, because they can never be interactive. Well, IBM didn't make their computers to be interactive. Almost all publications dealing with IBM's hardware and software philosophy point out that the goal of the 360/370 is to maximize the efficiency of the computer, not to accommodate the user. IBM did not pull away from the competition so strongly because it sat around and did nothing, or because companies wanted to buy interactive graphics systems. IBM is quite into supporting its stuff, which is more than can be said for 90% of the companies in the personal computing market. Would you want to depend on a company staffed by two people out in California to support a computer on which you want to write paychecks for thousands of people? Do you need dingdongs and doorbells to do mailing lists? OK,

commercial applications are not the ultimate applications, but they are necessary. Just because a Mack Truck isn't a Porsche doesn't make it wrong.

Nelson goes on to suggest that IBM should be quaking in its boots because of the personal computing revolution, which seems rather unlikely. As a matter of fact, the reviewer and another member of the staff of *Creative* have a running joke about the comparison between an Altair 8800 and a 370/168 (and NOT because we'd rather have an Altair.) IBM has no need to fear the time when the equivalent of System 370 hardware can be had for $9.95. Imagine, if that can be done for $9.95, what a few million will buy! Besides, the cost of developing computer software will *not* continue to fall. To date there have been only a handful of successfully mass-marketed software products (Tom Pittman's *Tiny BASIC* and the SWTPC 6800 BASICs come to mind, but they surely aren't high-powered exotic software tools.) In selecting IBM alone of all the big computer companies to attack, Ted Nelson is doing the computing community a disservice. It is possible to knock anything, regardless of its actual worth. I should mention at this point that the reviewer is *not* an IBM freak, though he has come to respect the power and flexibility of IBM equipment, as well as its incomprehensibility.

Ted Nelson also discusses some very nifty software tools, such as TRAC, SMALLTALK, LISP, and APL. Too bad you can't run down to your local computer store and buy any of those languages for any price. Ooops, forgot to tell you, TRAC is a registered trademark and servicemark of Rockford Research, Inc., Cambridge, Massachusetts. Although an 8080 version of TRAC is in existence, it isn't available to Joe Computer User, though Ted Nelson has one and enjoys it.

I regret that this has been a bit more of a rebuttal than a review, but someone has to speak up. I would certainly recommend *The Home Computer Revolution* to anyone involved with personal computing. For all its faults, it is very interesting and readable. As a matter of fact, it is the most controversial book on personal computing that has come along in awhile.

One hopes that the slick appearance of this book (in contrast with *Computer Lib*) does not indicate that the entire personal computing field will turn into a mass marketing phenomenon.

(Sorry Ted, but they can't all be gems.)

Steve North

[Readers who want to judge the book for themselves can order a copy from *Creative Computing* for the unheard-of low price of $2.00.]

*****

The First Book of KIM. Jim Butterfield, Stan Ockers, Eric Rehnke, editors. ORB, P.O. Box 311, Argonne, IL 60439. 176 pp., paperback $9.00. 1977.

The book is "dedicated to the person who just purchased a KIM-1 and doesn't know what to do with it..." Much of the material in the book has been taken from *KIM-1/6502 User Notes.* Material is collected under the following titles:

A Beginners Guide to KIM Programming, which takes a first time user through the steps of getting the KIM-1 to respond to one's commands. This section is brief but well-written and the user finishes with the confidence he shall be the master over the computer. This chapter assumes one has the *KIM Programming Manual* to be read as one becomes familiar with the operation of the KIM-1.

Recreational Games is just that, a series of ready to enter games, exercises and educational programs. Each program consists of an explanation or purpose, an assembled op-code listing (well documented) and a hex dump. The experienced user has only to sit down, plug in the KIM-1 and begin entering the code and in less than 15 minutes will become deeply involved in the variety of games in this chapter.

Diagnostic & Utility Programs consists of a series of programs that add flexibility to the KIM-1 system hardware and software system.

Expansion, Interface and Pot Pourri chapters collect together in one place those facts one tends to gather and lose when the information is needed.

The book is well-written, well-documented and highly recommended for all KIM-1 users, whether beginners or old-timers. With some difficulty these programs can be adapted to other 6502 systems using a monitor other than KIM.

John Jackobs
Cedar Rapids, IA

ws... reviews... revi

Fun With Computers and Basic. Donald D. Spencer. Camelot Publishing Company, P.O. Box 1357, Ormand Beach, Florida 32074. 96 pp., paper. $6.95. 1977.

This book is divided into three main sections. The first contains some common uses of computers and an introduction to some important computer related terms. The next section discusses BASIC. This brief treatment of BASIC includes the elementary commands thru FOR-NEXT, subscripted variables, and the library functions. The final part of the book consists of 64 pages of games, simulations, puzzles, and topics from number theory. Each problem is explained well, and over half are accompanied by the resulting program.

The author suggests that the book could be used by two main groups of people. The first includes personal-computer users or students who must learn BASIC on their own. The second group consists of teachers, students, programmers, and others who enjoy computerized game playing. I feel that the book lends itself nicely to use by the second group, but would miss the mark as a self-teaching text. I don't feel that a thirteen-page treatment of BASIC is quite enough. The explanation of some of the commands and functions is sketchy. For instance, the greatest integer function is explained as follows: "The INT function is used to 'chop-off' the fractional part of a value, resulting in an integer value." The example shown was INT(239.52) = 239. Won't the reader be surprised when he finds the value of INT(-8.29)?

Although the book is a paperback, the pages are sewn in and it appears to be of excellent quality. The type is large and very readable. Almost every page contains a diagram or cartoon related to the material being presented.

I would highly recommend this book as a source of fine problems for a high-school course in computer programming. It should be included in a teacher's personal library; it would be an asset to the school library; and it would make a great gift to a student who has already been exposed to BASIC.

Bruce De Young
West Milford, NJ

*****

Data Processing for Business, Second Edition. Gerald A. Silver and Joan B. Silver. Harcourt Brace Jovanovich, Inc., New York, NY. 596 pages. Hardbound. 1977.

A contemporary view of computer technology and computer languages is presented, moving from simple to complex in concepts, terminology, and theory. As in the previous edition, cartoons and anecdotes are used to present ideas, thoughts, and commentaries of people with a broad base of the computer as a social tool. Materials have been revised and updated to include such items as: point-of-sale terminals, electronic funds transfer system, floppy disks, microprocessors, and legislation on social implications of the computer.

The book is divided into 23 chapters organized into seven parts with four appendices, glossary, and index. Part 1 (chapters 1 and 2) gives an overview of data processing with terminology and trends in hardware and software. Part 2 (chapters 3 and 4) covers the concepts of the punched card and unit record machines. This area has been condensed from the previous edition. Part 3 (chapters 5-10) considers the hardware area with input, data representation (numbering systems), central processing unit, storage, and output. Part 4 (chapters 11-13) deals with the solving of problems with a computer; thus going through program planning, flowcharting, and processing methods. Part 5 (chapter 14-18) deals with software individual chapters on operating systems, COBOL, BASIC, and FORTRAN and a combination chapter touching on assembler, PL/1, RPG, APL, and ATS. The chapter on the BASIC language is new to the second edition, expanding from just three pages (overview) to a 19-page chapter. Part 6 (chapters 19-22) explores the areas of business systems, performance evaluation of systems, teleprocessing, and a new chapter on information systems. Part 7 consists of just chapter 23 but probably brings out the most important aspect — the computer

in society: impact on society. Topics covered in the chapter include; employment, industrial computer monopolies, new crimes, cashless society, and impact of data banks. This chapter probably should be placed in front of the book for more impact. The appendices list employment opportunities and job descriptions in data processing, conversion tables for numbering systems, keypunching procedures, and case problems. The case problems are excellent for class discussion to lead the student through applications with background information, problem, solution, and benefits.

The book is loaded with illustrations, cartoons, and stories which should help a novice understand data processing and its implications. Key words are printed in green to aid the reader know which words are important. Exercises at the end of the chapters help reinforce materials covered. A *Study Guide to Accompany Data Processing for Business* is available, covering each chapter with terms needed (repeat of textbook) and self-tests to measure comprehension. The tests consist of true/false, matching, and multiplechoice questions, and also included are several essay questions designed to guide the reader in synthesizing important concepts.

In comparison with the first edition, the second edition adds much and covers again historical to future concepts of data processing. Since the broad coverage of central topics excludes excessive details, the place for the book has to be a light approach to data-processing concepts. This fits the authors' design for use in a beginning course in data processing.

John F. Schrage
Fort Wayne, IN

*****

Electronics Sourcebook. Bill Prudhomme. Technical Publications, 1405 Richland Ave., Metarie LA 70001. 78 pp, paper. $3.50 (25¢ postage). 1977.

This little book tells how to get information and samples from electronics suppliers. Written in a gee-whiz style, it makes some straightforward suggestions about how to ask (type on letterhead rather than scribbling on lined paper and things like that), and gives a short bibliography of places to write. For those new to the electronics and computer field who are unaware of the considerable variety of information available mostly for free, this could be a useful introduction.

John Levine
New Haven, CT

*****

Stimulating Simulations. C.W. Engel, 64 pp. paper, $5. C.W. Engel, Box 16612, Tampa, Florida 33687.

Stimulating Simulations is a book containing ten "simulation programs," which are really game programs, though a few are also simulations. The programs were done in BASIC, apparently MITS/Microsoft 8080 BASIC, but conversion to other BASICs would be simple enough. Each of the programs is presented with a listing, sample run, instructions, and program documentation, including a flowchart and ideas for improvement. The programs are well written and are not rehashes of old programs. The following programs are in the book: Art Auction, Monster Chase, Lost Treasure, Gone Fishing, Space Flight, Forest Fire, Nautical Navigation, Business Management, Rare Birds, and Diamond Thief. The programs tend to be on the short side (under 100 lines).

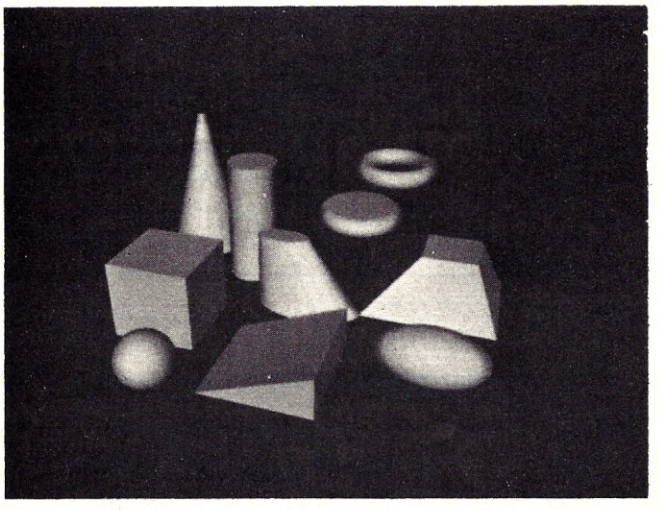
Many of the games/simulations also have potential educational application, such as the programs involving navigation which help teach the use of Cartesian coordinates, trigonometric principles, etc. Although the programs aren't too complicated, the excellent documentation would make it easy to modify the games yourself. And, as mentioned before, the programs in this book *are* original, so the book is quite worthwhile. The price of the book does seem rather high ($5) especially in comparison with other game books such as Creative's *BASIC Games,* or PCC's *What to Do After You Hit Return.* This is partially because *Stimulating Simulations* is a homebrew-type effort. Anyway, how many different versions of Star Trek do you want?

Steve North

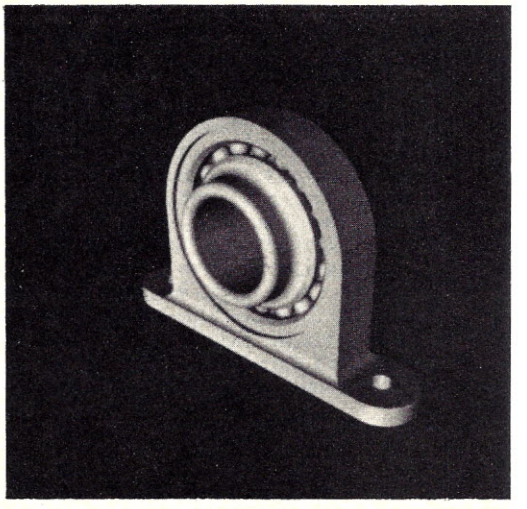
SYNTHAVISION

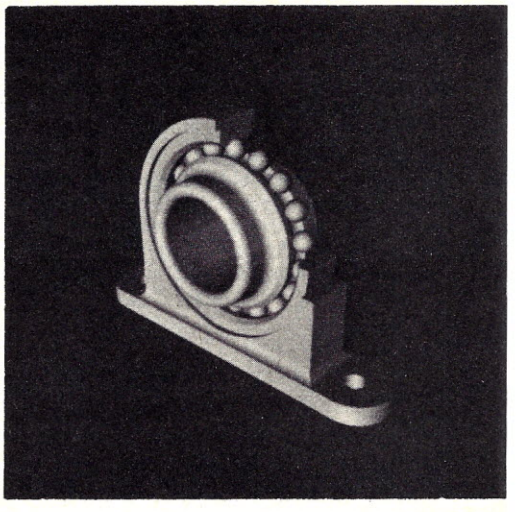
The building-block approach to 3D computer graphics

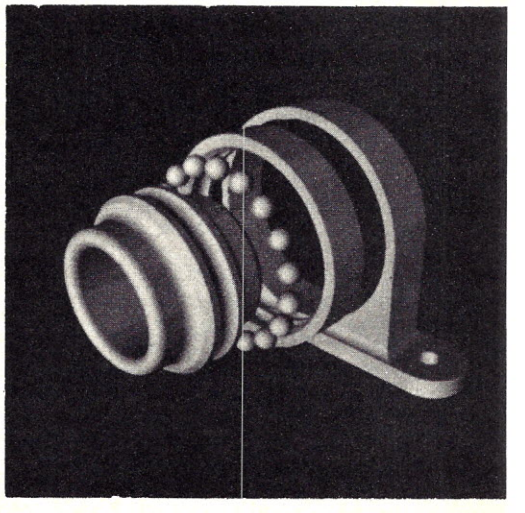
Larry Elin



Pictured here are ten primitives used as building blocks in the SynthaVision process. From left to right they are: sphere, box, cone, cylinder, wedge, elliptical cone, solid torus, torus, arbitrary polyhedron, and ellipsoid.







In this series of photos you see first a complete ball-bearing assembly, then the assembly with a box subtracted from one section to create a cross-sectional view, then the components separated to create an "exploded" view.

SynthaVision is a computer animation process that allows the producer/animator a great deal of flexibility and control in creating images of fully-shaded, three-dimensional objects. In a short time, the animator can describe very complex objects to the computer, input movement commands and colors, and leave the dog work to the computer.

The computer, in our case an IBM 360/75, spends an average of three minutes generating an image. This is a short time compared to traditional hand animation, but a long time as computer imaging systems go..

Purists would argue that computer animation systems should be "real time," avoiding a definition of "real time" but noting that three minutes per frame is too long. What they fail to realize is that for the animator and film maker in general, there never has been real time. Nor has there ever been a need for the immediacy implied by the term. Let's face it, for the video people, real time is instantly; for those of us who astral-project, it's even sooner than that; and for the home photographer, the closest thing to real time is the photomat store. It's all relative.

Larry Elin, Mathematical Applications Group, Inc., 3 Westchester Plaza, Elmsford, NY 10523

All computer animation systems that attempt to generate fully-shaded, three-dimensional images have to deal with the problem of describing surfaces to the computer. Most existing systems rely heavily on inputting a large number of point locations (x,y,z) that lie on the surface of the object they are describing. The computer then connects these points with polygon patches, other subroutines do smoothing and shading, and the resulting image resembles the object described.

The SynthaVision approach is quite different. Certain three-dimensional primitive shapes are *already* programmed into the computer. These shapes are solid volumes such as box, sphere, cone, ellipsoid, elliptical cone, wedge, torus (both with and without a hole) and cylinder. These shapes are added to or subtracted from one another to form a more complex shape.

Virtually, anything man-made can be described using these simple shapes as building blocks. In fact, the process of using primitives to build complex objects is not unlike the thinking process that a design engineer goes through when he or she conceptualizes a part for a car or a machine. The chief advantage of the SynthaVision process is that you don't have to have the real object on hand in order to describe it. You can, in effect, make one up!

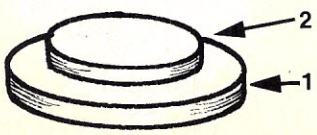
For example, if you want to describe a simple ashtray, you would need at least two geometric primitives, both of them cylinders. The first step is to describe the cylinder that will be the solid bottom and sides of the ashtray. You decide what its dimensions will be, and input them thus:

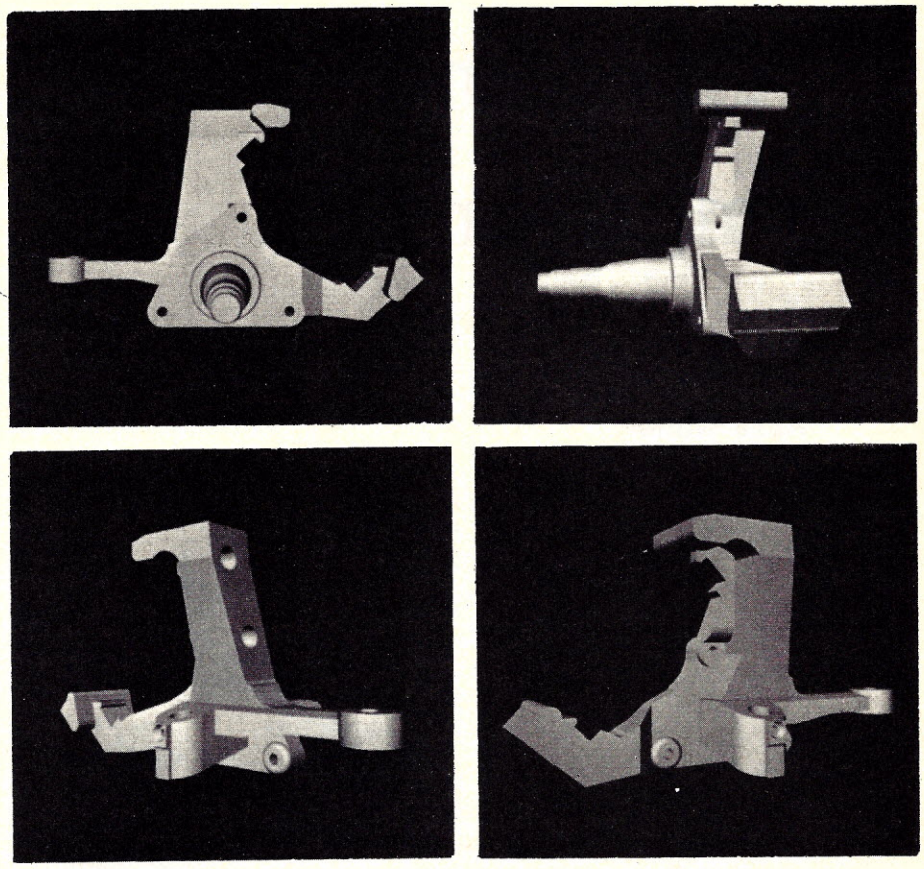
```
RCC 1  0.  0.  0.  0.  2.  0.
       5.
```

This data would be interpreted by the computer as meaning Right Circular Cylinder #1 is located at X0, Y0, and Z0; it is 2 units high in the Y direction and has a radius of 5. units. Then you would describe another cylinder whose location and dimensions are:

```
RCC 2.  0  1.  0  0  2.  0
        4.
```

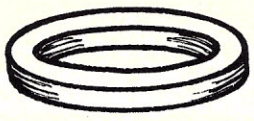
This puts RCC 2 a little higher and a little smaller than RCC 1:





Here you see an 18-mm spindle for the wheel of a car which, once described, can be rotated about a central axis to show other angles. In this case, the spindle was rotated 70 degrees for each picture.

If you then subtract #2 from #1, you would wind up with:



Naturally, this is a very simple example, but you get the idea. The addition and subtraction can go on and on until you have a very complex object.

Simple English-language instructions are used to animate the object once it is described. A typical command is:

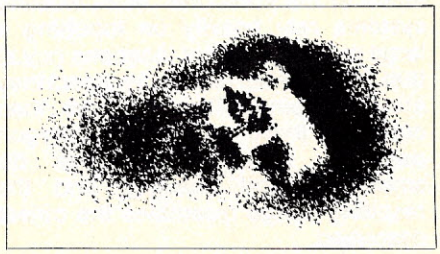
```
1  50  Move TRAY    2.  0.  0.
```

This command means: from frames 1 to 50, move the object named "tray" 2 units in the x direction, 0 units in the y and 0 units in the z.

The computer calculates what that object would look like if it moved 2. units in each frame, and outputs each picture on a Cathode Ray Tube. A movie camera pointed at the CRT photographs the image as it is displayed.

So far, SynthaVision has been used to produce over 200 commercials, educational and industrial films. It is especially useful for describing how something works. After trying to write this article, I'm convinced we should use it to make a film showing how SynthaVision works!! ■

Creating Computer Art



DIFFUSED KENNEDY

Russ Walter

SHOT KENNEDY

KENNEDY IN A DOG

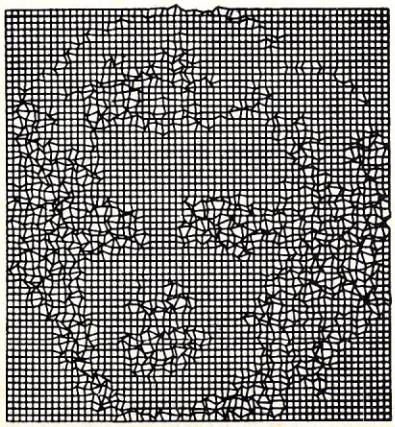
CREATING ART

Every black-and-white photograph can be expressed as a table of numbers. Each number in the table represents the darkness of a different point—the higher the number the darker the point. The "darkness numbers" are called *gray levels*. To feed a picture into the computer, type in the table of gray levels. Or aim a special camera (called an *optical scanner*) at the object you want pictured; the scanner will automatically compute the gray levels and send them to the computer via a wire.

You can program the computer to change the gray levels in any weird way you wish, and draw the result. Here's what the Computer Technique Group of Japan did to an ordinary photograph of John Kennedy:

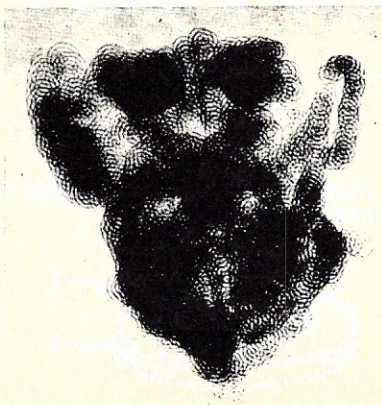
Here's what the group did to a photograph of Marilyn Monroe:

MONROE IN THE NET



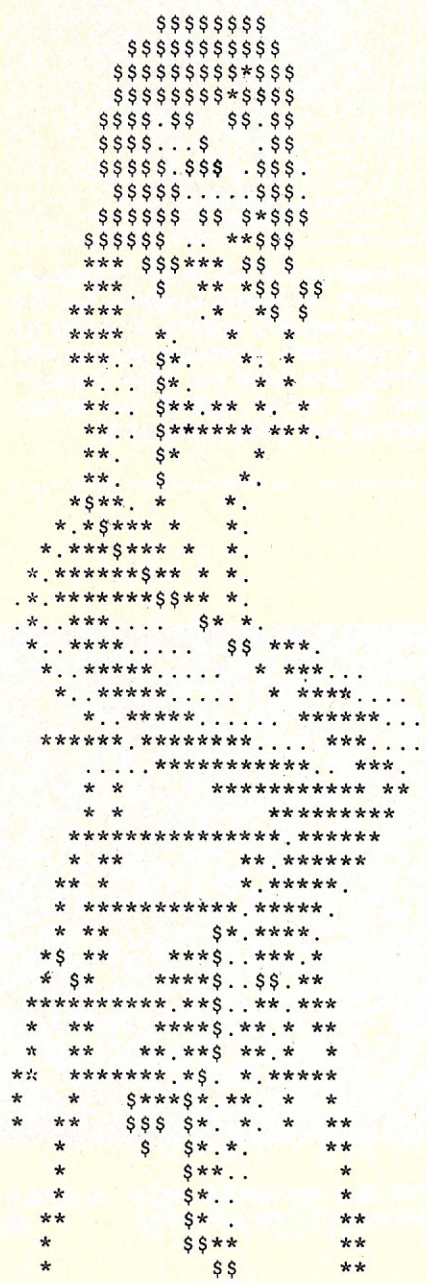
Csuri & Shaffer fed the computer a realistic line drawing of an old man; here's what came out:

RANDOM LIGHT AND SHADOW



I did this with the help of a computer:

PIN-UP



The *Pin-Up* has these specifications....

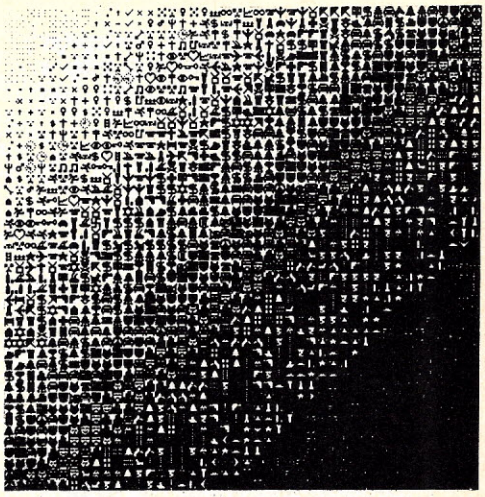
scene: a scantily clad girl sitting on a stool - 4 different gray levels - 4 different symbols (1 for each gray level): a blank, a period, an asterisk, and $. 1537 symbols altogether (53 rows x 29 columns)

In the specifications, the numbers are small, yet the picture is clear. The clarity was obtained by non-computerized fanagling.

At Bell Telephone Laboratories, Knowlton & Harmon produced a picture with much larger specifications....

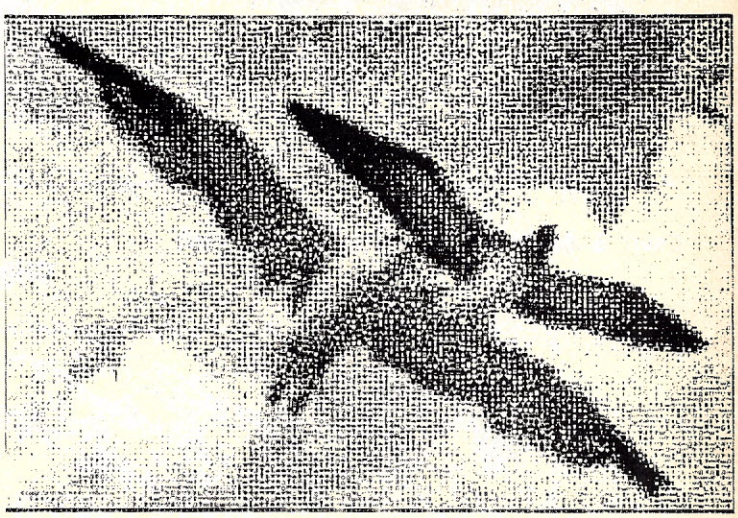
scene: two seagulls flying in the clouds - 16 different gray levels - 141 different symbols (for each gray level there are several symbols; the computer chooses among them at random). 11616 symbols altogether (88 rows x 132 columns)

Instead of using blanks, periods, asterisks, and $, they used cats, battleships, swastikas, and other weird shapes. Here are the 141, listed from lightest to darkest, with some repetitions:

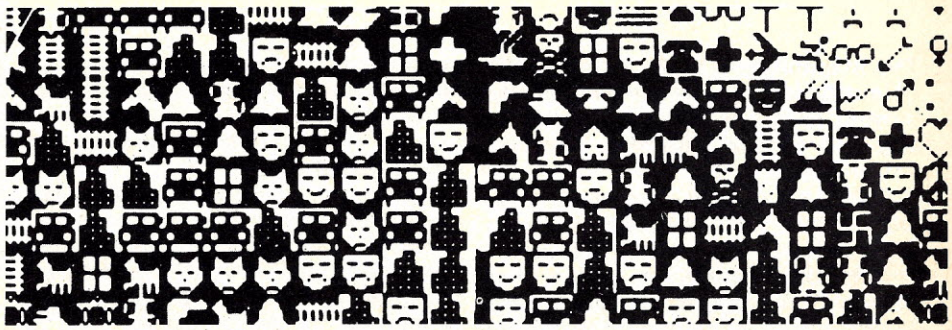


GULLS

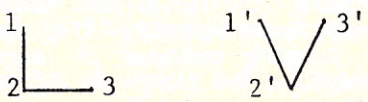
The picture is several feet long. Seen from a distance, it looks like this:



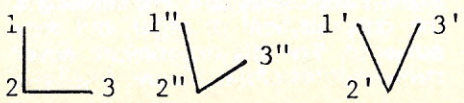
Here's a close-up view of part of one of the gull's wings:



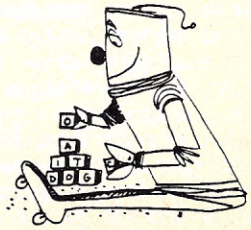
Here's how to make an L slowly become a V. Notice that the letters L and V are both made by connecting three points:



Let 1″ be the point halfway between 1 and 1′; let 2″ be halfway between 2 and 2′; and let 3″ be halfway between 3 and 3′. Then 1″, 2″, and 3″ form a shape that's halfway between an L and a V:

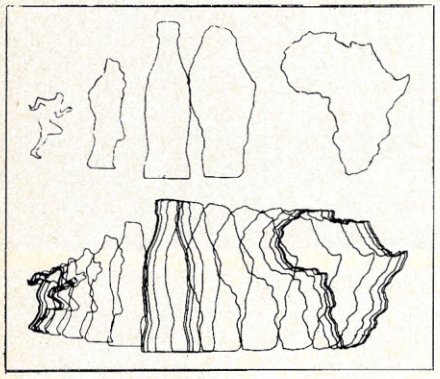


The process can be extended further:



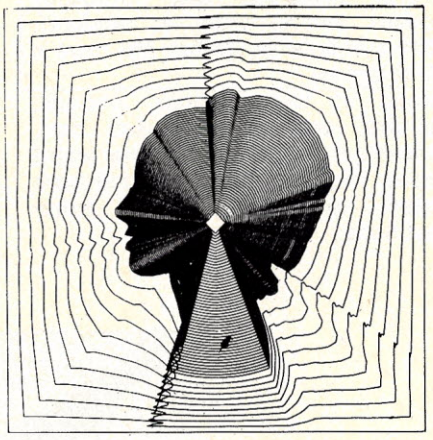
Using that method, the Computer Technique Group gradually turned a running man into a Coke bottle, and then into Africa:

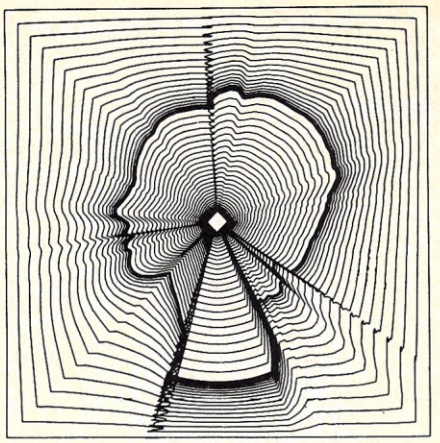
RUNNING COLA IS AFRICA



The group turned this head into a square:

RETURN TO A SQUARE





The head on the left returns to a square by using *arithmetic progression:* the lines are equally spaced. The one on the right uses *geometric progression* instead: the lines are close together near the inside square, but further apart as they expand outward.

Csuri & Shaffer exploded a hummingbird:

CHAOS TO ORDER

The hummingbird at the far right was obtained from the one at the far left, by moving each line a random distance and in a random direction (between 45° and -45°).

Computer artists are starting to believe that *art is a tension between order and disorder.* Too much order, or too much disorder, is boring. For example, in *Chaos to Order,* the hummingbird on the left is too orderly to be art. The hummingbird on the right is more interesting.

Consider *Gulls* (page 85). Seen from a distance, it is an orderly picture of gulls. Seen up close, it is an orderly picture of a cat or a battleship or a swastika. But from a middling distance, it looks like disorderly wallpaper—the symbols repeat, but not in any obvious cycle. That element of disorder is what makes the picture interesting.

At first glance, *Pin-Up* (page 85) is just a disorderly array of periods, asterisks, and dollar signs. At second glance, you see order: a girl. *Art is the formation of order from disorder.*

A first glance at *Monroe in the Net* (page 84) shows order: a piece of graph paper. A second glance shows disorder: some of the graph's lines are inexplicably bent. A third glance shows order: Marilyn Monroe's face pops out at you. Her orderly face is formed from the disorder of bent lines.

Return to a Square (page 86) uses arithmetic progression and geometric progression to create an over-all sense of order, but the basic elements are disorderly: a head that's bumpy, and a panorama of weird shapes that lie uncomfortably between being heads and squares but are neither.

Many programs create disorder by random numbers. *Chaos to Order* uses random numbers to explode the hummingbird. *Gulls* uses random numbers to help choose among the 141 symbols. An amazing example of random numbers is this picture by Julesz & Bosche:

To your eyes, the picture seems quite ordered. Actually, it is quite *dis*ordered. One pie-shaped eighth of it is entirely random; the other seven eighths are copies of it. The copying is the only element of order, but very powerful. Try this experiment: *cover seven eighths of the picture.* You'll see that the remaining eighth is totally disordered, hence boring. ■

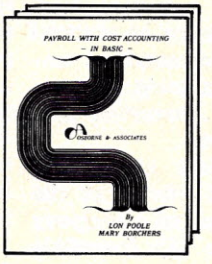
OSBORNE & ASSOCIATES, INC.

The World Leaders In Microprocessor Books

If you want information on microprocessors, begin with the Osborne books.

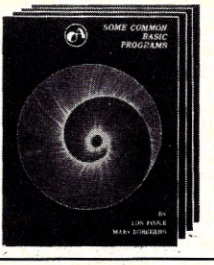
PROGRAM BOOKS WRITTEN IN BASIC

Payroll With Cost Accounting
Accounts Payable And Accounts Receivable
General Ledger



These books may be used independently, or implemented together as a complete accounting system. Each contains program listings, user's manual and thorough documentation. Written in an extended version of BASIC.
#22002 (400 pages), #23002*, #24002*

Some Common BASIC Programs



76 short practical programs, most of which can be used on any microcomputer with any version of BASIC. Complete with program descriptions, listings, remarks and examples.
#21002 (200 pages)

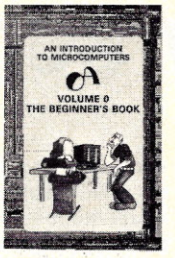
ASSEMBLY LANGUAGE PROGRAMMING

8080A/8085 Assembly Language Programming

6800 Assembly Language Programming

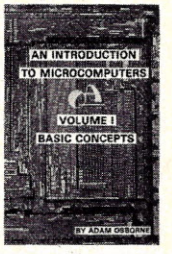
These books describe how to program a microcomputer using assembly language. They discuss classical programming techniques, and contain simplified programming examples relevant to today's microcomputer applications.
#31003, 32003 (400 pages each)

AN INTRODUCTION TO MICROCOMPUTERS



Volume 0 - The Beginner's Book

If you know nothing about computers, then this is the book for you. It introduces computer logic and terminology in language a beginner can understand. Computer software, hardware and component parts are described, and simple explanations are given for how they work. Text is supplemented with creative illustrations and numerous photographs. Volume 0 prepares the novice for Volume I.
#6001 (300 pages)



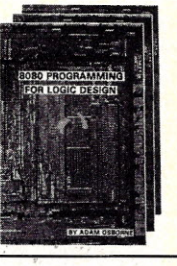
Volume I — Basic Concepts

This best selling text describes hardware and programming concepts common to all microprocessors. These concepts are explained clearly and thoroughly, beginning at an elementary level. Worldwide, Volume I has a greater yearly sales volume than any other computer text.
#2001 (350 pages)

Volume II — Some Real Products (revised June 1977)

Every common microprocessor and all support devices are described. Only data sheets are copied from manufacturers. Major chip slice products are also discussed.
#3001A (1250 pages)

PROGRAMMING FOR LOGIC DESIGN



8080 Programming For Logic Design
6800 Programming For Logic Design
Z80 Programming For Logic Design

These books describe the meeting ground of programmers and logic designers; written for both, they provide detailed examples to illustrate effective usage of microprocessors in traditional digital applications.
#4001, #5001, #7001 (300 pages each)

OSBORNE & ASSOCIATES, INC. • P.O. Box 2036 • DEPT. L3 • Berkeley, California 94702 • (415) 548-2805 9:00 a.m. - 5:00 p.m. PST
TWX 910-366-7277

Price applies to orders received by June 30, 1978.		PRICE	QTY	AMT
6001	Volume 0 — The Beginner's Book	$ 7.50		
2001	Volume I — Basic Concepts	7.50		
3001A	Volume II — Some Real Products (1977 edition)	15.00		
4001	8080 Programming For Logic Design	7.50		
5001	6800 Programming For Logic Design	7.50		
7001	Z80 Programming For Logic Design	7.50		
31003	8080A/8085 Assembly Language Programming	7.50		
32003	6800 Assembly Language Programming	7.50		
21002	Some Common BASIC Programs	7.50		
22002	Payroll With Cost Accounting	12.50		

- 6-1/2%, SF Bay Area residents only
- 6%, California residents outside SF Bay Area
- Payment must be enclosed for orders of 10 books or less.
- I have enclosed:

☐ check ☐ money order

TOTAL
Sales Tax (Calif. residents only)
Shipping Charges
TOTAL AMOUNT OF PURCHASE

NAME

ADDRESS

CITY STATE ZIP PHONE

SHIPPING CHARGES Shipping charges for bulk orders to be arranged.

☐ 4th class (no charge, allow 3-4 weeks within USA, not applicable to discounted orders)

☐ $.50 per book, UPS (allow 10 days) in the U.S.

☐ $1.50 per book, special rush shipment by air in the U.S.

☐ All foreign orders, $3.00 per book, for air shipment

***These books are scheduled to be published during 1978.**
Please notify me when they are available:
☐ 23002 Accounts Payable and Accounts Receivable
☐ 24002 General Ledger

Please send information on:
☐ pricing, ordering, and titles available after 6/30/78
☐ initial dealer consignments
☐ dealer and school discounts
☐ foreign distributors

Price increase scheduled for July 1, 1978

F8

CIRCLE 118 ON READER SERVICE CARD

Computer animation: Out of the lab and onto the tv screen

BY ARTHUR BELLAIRE

Are television commercials looking and sounding more and more alike? Or does it just seem that way?

From a technique standpoint, how many party situations, cookouts, fun on the beach, quick cuts to happy faces, slices-of-life, pitchpeople can a viewer absorb in a single evening? Individually, many are right on and, without a doubt, highly effective. But once they leave the clients' screening rooms and mix into the swirl and clutter of the real world, how many others become blurred and forgotten before the next station break? Even many spots in the new bury-your-competitor school of comparison and advertising risk losing more in confusion and sameness of look than gaining in persuasion.

We once considered technique secondary to basic message—still a healthy way to think. But what good is a basic message if some kind of distinctive vehicle doesn't drive it deeply and firmly into the viewer's mixed-up brain? With heavily advertised products such as foods, beverages, drugs and toiletries, *how* you show it and say it on television *has* to border on basic, because here is the tap on the shoulder that determines whether or not your commercial will be noticed at all, let alone absorbed.

Which leads to Question No. 2: To what extent, then, is the success of a product advertised on tv due to sheer weight of media dollars as compared with message and technique? We may never know.

Computers Can Solve Creative Problems

What we do know is that in television we have a medium of limitless technological possibilities to depict every kind of scene and symbol the human mind can possibly comprehend. The question is, after this first quarter-century of using the medium to move products and services, how far beyond the obvious have we really explored into the myriad hidden opportunities which may help us motivate with pictures? This piece concerns one such possibility which has recently come of age—computer animation. Never to be considered a replacement for solid and proven live action, computer animation is nevertheless perfect and available for those special times when an unusual creative problem may demand an unusual solution. Or when a going, successful campaign needs a change-of-pace variation to sustain interest.

Allan Stanley, president of Dolphin Productions, biggest and busiest of the computer animation firms and a company that has helped advance this incredible technique to sophisticated proportions, prefers to call the process electronically-generated animation, which in no way replaces conventional cel animation, but carries its range of graphics forms into unlimited dimensions.

For the technically oriented, the Dolphin system offers the unique capability of animation in real time. The output is in standard color video available for immediate use and/or integration into other color video material. The process is completely interactive, permitting constant esthetic evaluation and direction while the motion and sequence dynamics are being set. The input is artist-prepared graphics, drawings, charts, etc. Because the system is "real time," it can be instantaneously mixed with standard color video signals, including live action, coming from studio color cameras, pre-recorded videotape and a duplicate animation system.

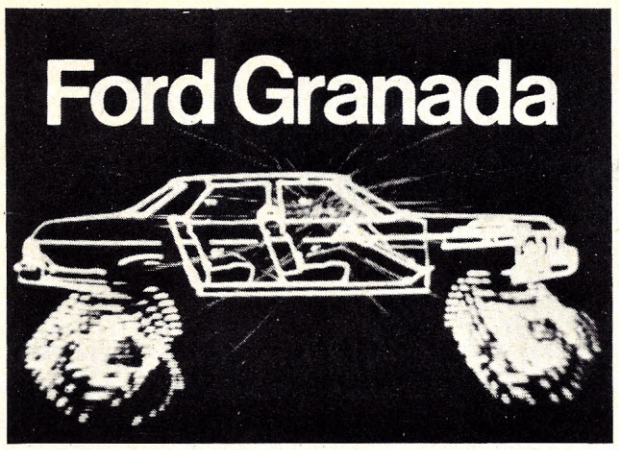
■ Or for the layman like me, what happens on the screen is figures twist, squeeze, stretch, zoom in or out, strobe against three-dimensional fields. "We have the ability," adds Mr. Stanley, "to create motion on any element of a picture independently, even to colorize any of these elements while in motion independently." The motion can happen at any speed. Backgrounds can be bursting stars, explosions of dots—you name it. An expensive, Hollywood-type set can be simulated through keying behind a live singer. All this happening as experts at the 15-ft.-long console push the right buttons and plug in and unplug the right lines. And it all comes out either on two-inch videotape or 35mm film. Your choice.

Many of these effects, if attempted optically and with cel animation, would be prohibitive in price. Yet Dolphin delivers the final commercial within two days (if on tape) at a cost to the advertiser, they say, about one-half the price of the average commercial.

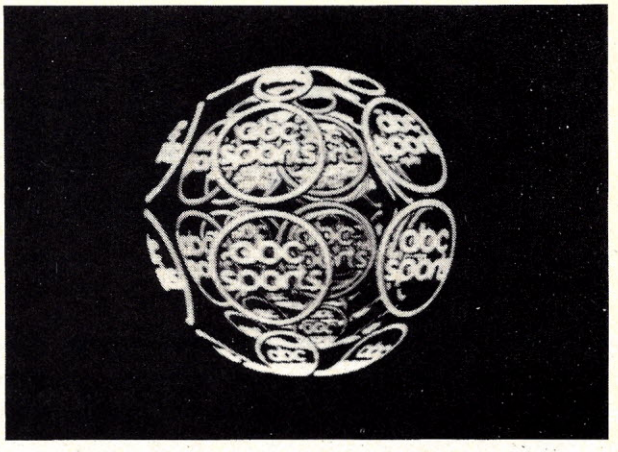
Wild Graphics Teach Children to Count

Fifty per cent of the studio's output relates to commercials. The remainder goes to networks and stations for the advertising of programs, program titles, station ID packages, in-company training films and pr presentations, sales and motivational shows. It recently contracted with a state educational department for teaching the new metric system. And, for Children's Television Workshop, here's where most of those wild and intriguing graphics come from in teaching our kids how to

"Mame" ad uses rhythmic interplay of computer-animated artwork and musical score.



Different parts of new Granada join together in seconds.



Computer animation brings vitality to ABC Sports logo.

count and spell. For these various contributions, the studio is already sporting 27 national and international awards.

A Dolphin commercial for Ford Granada, through J. Walter Thompson, features the engineering design of the new car, with different parts of the car joining together into the complete unit within seconds—all from a single piece of artwork—and positions the car against a glamorous background.

The studio worked entirely from Bob Peak's print ad artwork to bring into motion spectacular computer-animated commercials for the movie version of "Mame." Working against the "Mame" musical track, the various still elements interplayed rhythmically, catching a degree of excitement more conventional methods could not have matched.

Dolphin's computer animation has also brought new meaning and life to once-static corporate identities such as logos and slogans, zooming them into motion, often swirling behind, in front of, and around them intriguing symbols to impart consciously or subconsciously positive new moods and impressions. And when we notice sadly the too-quick handling so many packages receive in the final three seconds of so many product commercials these days, it's interesting to imagine whether any added sales may have resulted had the package been allowed, say, six seconds at the end in which to build, to move, and suddenly (while remaining literal) to become part of a totally different kind of scene. For Eveready batteries (William Esty/Film-Fair) a series of flashing, electric-like symbols fashioned through the computer animation technique suggested far more than just a battery with a name on it.

Dolphin's five-story townhouse headquarters on New York's upper east side contains just about every kind of video equipment you'll ever see in one townhouse. "But any company can buy equipment," says Stanley. "What's important are the people. We can prepare original artwork and have our creative staff take it from initial concept and storyboard to completion and delivery. We also welcome the creative input of agencies where, working together, we can program the various images into a sequence of motion that can be endlessly repeated and refined until complete satisfaction is attained before it is stored. As one of the agency people put it, 'It's discovering and participating in a new level of creative expression'."

Fun or not, from the standpoint of pure, no-nonsense advertising value, is there more here than meets the senses? Are such fantastic mixtures of symbols with still photos, artwork and/or live action (or just the symbols by themselves) one answer—even when used on an intermittent, change-of-pace basis—to the problem of sameness and boredom?

Symbols vs. Literal Presentation

More important, is it possible that the right combination of symbols, whatever they may be, integrated into the selling act within a commercial, could stimulate the viewer into accepting more of a selling proposition than through live action alone, which spares her the chore of thinking? Why isn't it logical that at least a temporary injection of the symbolic may even heighten the motivating powers of the commercial over the literal presentation? Could be we worry too much about sparing our viewer the need to concentrate, and even help her mind go stale, when the more challenging mental exercise which the symbolic approach demands may be the very thing she is waiting for.

This is obviously beginning to get too deep for me, but there are a number of research geniuses making it big in this business grinding out scores on total commercial performance whose time may be more profitably occupied taking us all back to "go" and explaining, one more time, how the customer's mind really works.

The point is, no amount of enthusiasm for computer animation or any other offbeat graphic idea is intended to suggest the technique as a steady substitute for what we know is working—straightforward, down-to-earth, nose-to-nose, live-action commercials for live-action viewers. Yet here and there, from time to time, now and then, occasionally, why shouldn't a willingness to reach out and try unusual approaches be a constructive way to freshen up the commercial look? Perhaps we can learn something in the bargain about this dynamic advertising medium we may never fully understand.

Reprinted with permission from *Advertising Age*, June 2, 1975.

Flashing, electric symbols say Eveready better than ever.

The Goodyear blimp—that charming reminder of a bygone era—has joined the ranks of the electronic media.

SKY LIGHTING

By Marc Treib

You don't see many blimps around anymore. Actually, you never did. Although lighter-than-air-craft date to the Montgolfier brothers' balloon of 1783 and perhaps back even further, they reached their apogee in the twenties and thirties when huge dirigibles like the rigid-frame Hindenburg plied the skies of the Atlantic on trans-world journeys. But the explosion of the Hindenburg in 1937 at Lakehurst, N.J., brought the history of these airships to a resounding close. Almost. During the Second World War, a new breed of airship, the blimp, served and served well for escorting convoys and maintaining submarine surveillance. Lighter than air, with no rigid frame, filled with helium (instead of the explosive hydrogen of the dirigible era), it could travel relatively quietly, had a great range, and could hover. After the war, however, not too much was heard of or about even the blimps.

Today, according to Tom Riley, who travels with the 22-person Goodyear-Columbia blimp crew, there are six blimps operating in the world. Four of these, the "aerial ambassadors," are the products of the Goodyear Tire and Rubber Company of Akron, Ohio, and serve primarily as public relations vehicles both in their commanding form and the messages they broadcast. One of the remaining two is located in West Germany where it serves as promotion vehicle for a beer company, and the last is in Japan. The difference in purpose is worth noting.

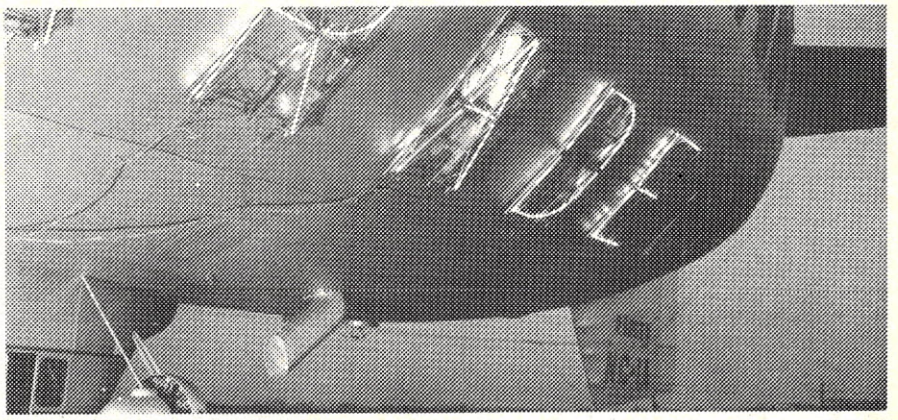
The first promotion or advertising that the blimps displayed almost 45 years ago was the lettering and winged foot of the Goodyear logo. Later on, came banners up to 200 feet long, trailing behind the blimp like those behind biplanes which still frequent beaches inducing watchers to use certain suntan lotions or go see a certain attraction. In a press release, Goodyear even mentions that in "the 1930s a loudspeaker was attached to the airship to permit voice contact with persons on the ground. The practice was short-lived, however, because people didn't like the 'voice from the sky.' "

Later on, neon tubing was used, carried by special lightweight metal frames and fixed to the sides of the blimp. "Known as 'Neon-o-gram,' this sign had the neon tubing shaped in such a manner as to permit the formation of any number or letter. Each sign consisted of ten individual frames." These sight-boards were used until the Second World War. After the war, frames consisting of individual bulbs, in the manner of the Trans-Lux news sign devices, were used. The color was limited to one—white—while the image was limited to basic messages. It was a major jump to the "skytacular" signs of today.

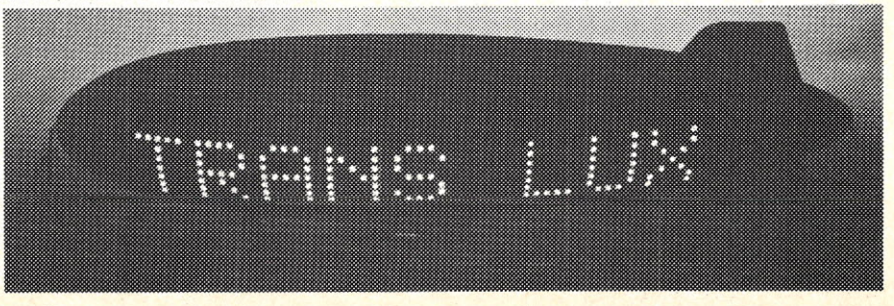
In the mid-sixties, under the instigation of Goodyear's vice president of public relations, Robert H. Lane, the new program for the "skytacular" and "super-skytacular night-signs" was developed. The "skytacular" was to be a system by which a light board could produce and broadcast messages in color and animation, freed from the restrictions of monochrome and text. The first version was installed on the former Mayflower in 1966.

The "super-skytacular night-sign" featured on the Columbia is the most elaborate version of the concept. It features some 7,560.175-miliamp bulbs operating under a 28-volt system. The electricity is produced by a special jet turbine engine mounted in its own pod behind the cabin which produces the 500-amp 28 volts required for operating the sign. All the electronic equipment for running the messages is housed within the small gondola of the blimp itself, although it only broadcasts

Thirties blimp with neon tubing. Sign was called "Neon-o-gram."



Forties blimp had frames consisting of individual bulbs in the manner of Trans-Lux news signs.



pre-recorded tapes. The light bulbs are fixed directly to the sides of the blimp and measure approximately 25″ high by 105″ long. The lamps are mounted within a special reflector and have either a red, blue, yellow, or green filter to control the color, and are connected by almost 80 miles of wiring. The variability of the image is virtually limitless.

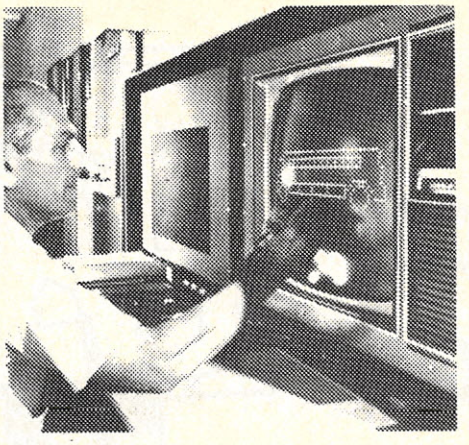
The Columbia is capable of carrying such animation as "a golfer driving his ball toward the green, then putting the final distance to the hole . . . a game of table tennis . . . a football player making good on a field goal . . . a sharp-shooting basketball player scoring on a foul shot and a long field goal attempt . . . a baseball player rapping a solid line drive, and almost taking off the pitcher's head in the process."

Special animated messages in color have been developed for the holidays. "Santa, his sleigh and reindeer flash across the Yuletide skies and the Magi and their camels follow the Star of Bethlehem . . . a turkey runs to escape an ax-wielding man intent on securing Thanksgiving dinner . . . a youngster lights a giant Fourth of July firecracker which explodes to form an American flag."

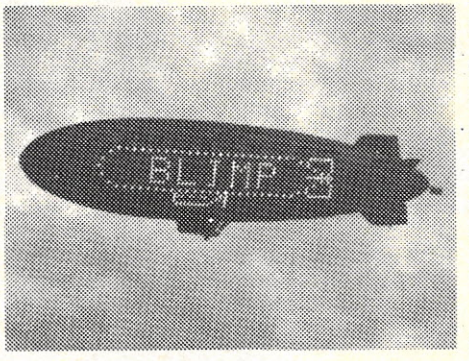
All the tapes which operate the signs are produced by the Night Sign Laboratory in Akron, home of the Goodyear company. A light pen is used to trace the designs on a cathode ray tube; the designs are then processed by the computer and recorded on magnetic data tape.

A typical six-minute tape consists of 40 million pieces or bits of "on-off" information which, when run through special solid-state electronic readers aboard the airship, control lamp and color selection and the speed at which messages are run.

These tapes are sent to the blimps for broadcasting. Although there is a Local Control Unit with each blimp, it is rarely used, except to produce limited local copy.



Night Sign Laboratory at Goodyear headquarters in Akron. A light pen traces night-sign designs on cathode ray tube. Designs are then processed by computer and recorded on magnetic data tape.



Typical night-signs.

For the most part, the word (and image) is delivered from Akron unto the blimp.

Although the possibilities for advertising via the blimp are almost limitless, about three-quarters of its messages are given over to public-service messages. Each year, the blimp is besieged by requests for blimpcasts ranging from appeals for Easter Seals, the Heart Fund and the Cancer Society, to birthday greeting requests. For the most part, broadcasts are limited to such messages as "Buckle seat belts," "Drive safely," and "Prevent forest fires." The remaining one-fourth of the time is reserved for advertising Goodyear products. But even then it's soft-sell.

The Columbia, which flies the West Coast,* divides its year in half. For six months it flies in the Los Angeles area or undergoes maintenance at a service facility in Santa Anna, where it is painted, the night-sign repaired, bulbs replaced, engines overhauled, etc. (A blimp's bag with no interior frame is replaced every five to seven years.) For the remaining six months, it takes to different skies. This year's tour, from June to November, included San Francisco, Portland, Spokane (for Expo '74), Seattle, and back to the San Francisco Bay Area in time to broadcast aerial views of football games during the fall season.

Have there been any complaints about violation of privacy from the skies? Surprisingly few, according to Tom Riley, And these are usually when the blimp is hovering for a game or special occasion. Much quieter, and less annoying, than helicopters, the blimp is usually regarded as a welcome and fascinating addition to the evening skies.

*The three other Goodyear blimps are the America, based in Houston, the Mayflower in Miami, and the Europa in Italy.

Color Graphics with a Light-Pen

Tom Dwyer

Margot Critchfield

One of the most popular exhibits at the 1977 Personal Computing Conference in Atlantic City was the color display demonstrated by the Computer Mart of New Jersey. It featured a "menu-selecting" program written by Bob Lindley for the Compucolor 8051 personal computer.

Fig. 1 shows a picture of output from the first part of Bob's program. The picture on the screen explains how to use a light-pen to input information to the computer program (written in BASIC). Fig. 2 shows a later part in the demo where the light-pen has been used to select both the colors (white, green, and red) and a drawing mode (continuous lines) to create a zig-zag "doodle." In Fig. 3, the "box" mode has been selected to create a more structured pattern consisting of rectangles of various dimensions.

Both colors and drawing modes are selected by pointing the light-pen to various "menu" positions around the border of the graphic drawing area. By picking up the X-Y coordinates of the pen, the program "knows" what part of the menu was selected.

Bob's program is rather long, and interested readers are advised to contact him at the Computer Mart. The principles behind use of the light-pen can be explained with a simpler program, however. We'll give one here, taken from the book *BASIC and the Personal Computer* (Addison-Wesley Co., Reading, MA 01867). Additional details about programming color graphics of this kind can be found in Chapter 10 of the book.

Using Light-Pens as Input Devices

A light-pen is a cigar-shaped device with a photo-cell at the tip, and a wired connection to the computer. When the photo-cell is held against the face of the CRT graphics display, the cell detects the light coming from a special "position-reporting pattern" on the screen. The X and Y coordinates of the position, being touched by the pen, are then sent to the computer. Fig. 4 illustrates the setup.

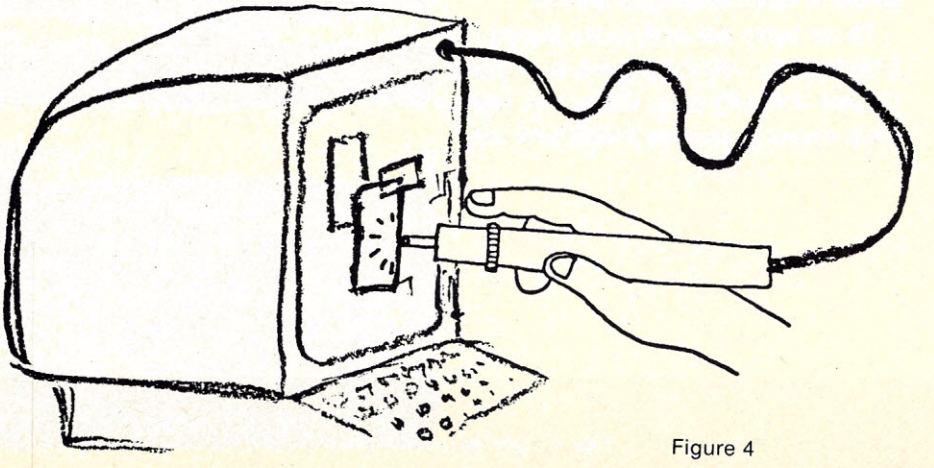


Figure 4

Tom Dwyer, Margot Critchfield, University of Pittsburgh, PA 15260.

Examples of Color Graphics with a Light Pen

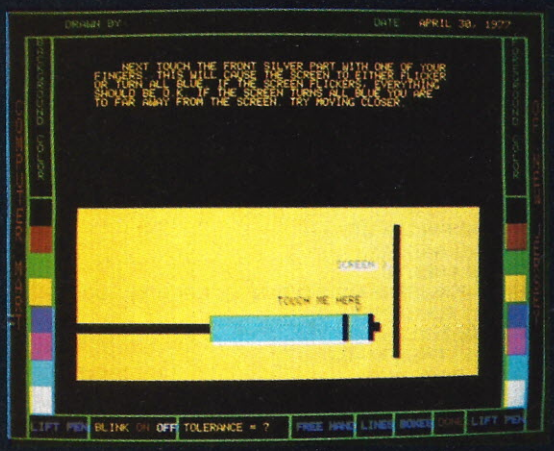


Figure 1. Instructions for using light pen is shown at beginning of the program.

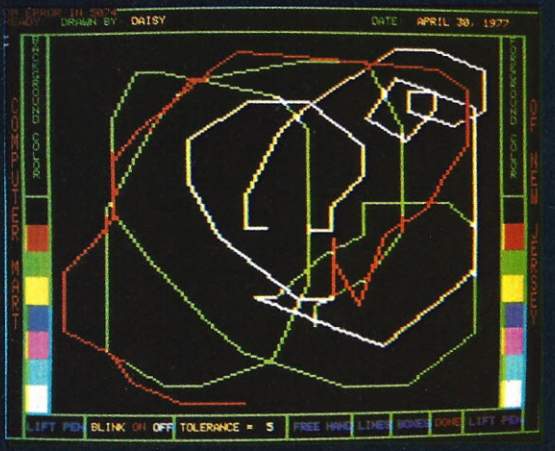


Figure 2. In drawing mode, the light pen can be used to select colors and draw a "doodle" or any other figure.

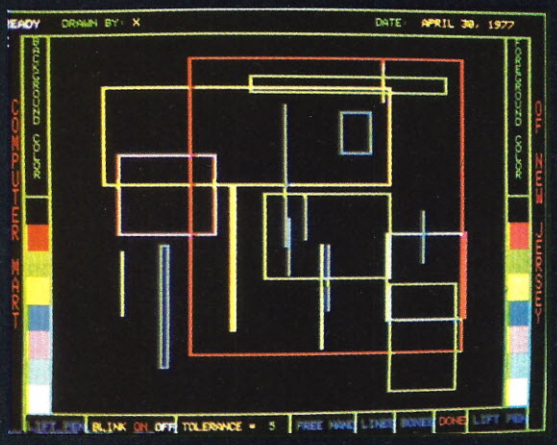


Figure 3. The "box" mode can be used to create a pattern of squares or rectangles.

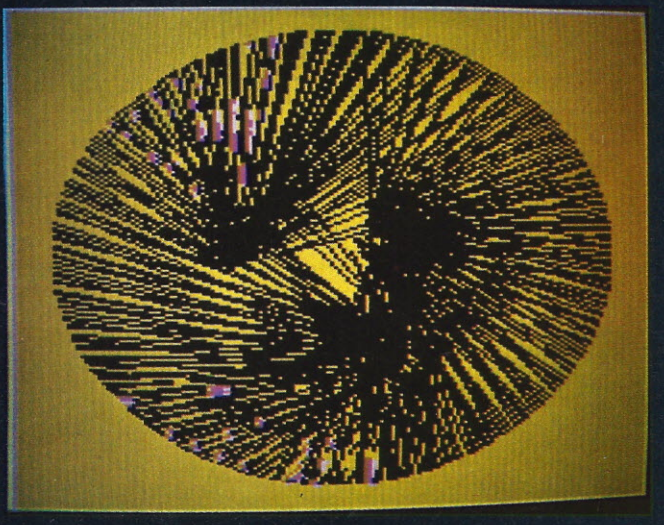


Figure 7. An infinite loop can be used to generate an ellipse or other shape.

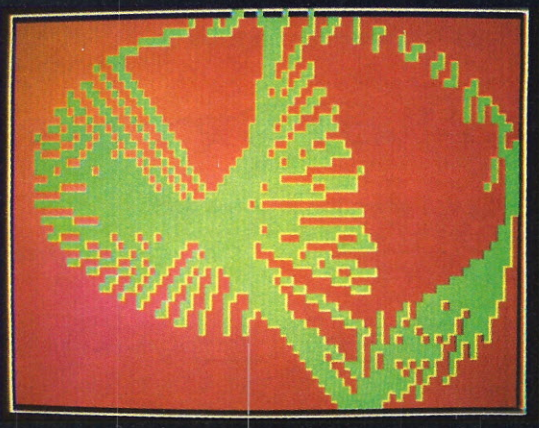


Figure 8. Another example of an infinite loop to generate a different ellipse.

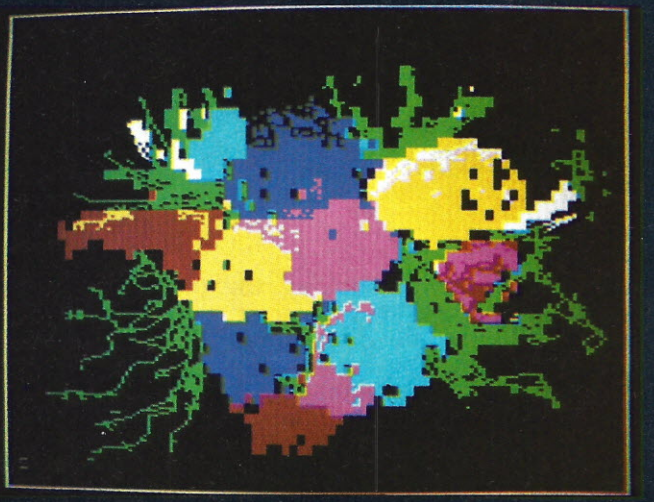
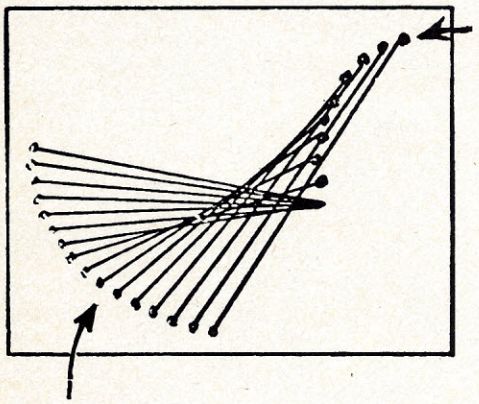


Figure 9. This bouquet of flowers was drawn using the light pen as an electronic paint brush.

This gives the user a very human-oriented way of providing X-Y input data to a program. To illustrate how this works, let's look at a BASIC program that uses the light-pen on the Compucolor computer. The pen is used to input the values of X1 and Y1. A vector is drawn from X0, Y0, to X1, Y1, where X0 and Y0 are values calculated within the program. As long as the pen is activated, the program continues to draw vectors from successively changing values of X0 and Y0 to whatever X1, Y1 is sent from the pen. If the pen is not activated, no new values of X1, Y1 are picked up from the pen and no values of X0, Y0 are calculated—the user can sit and think. Fig. 5 shows an example of output from the program.



X0, Y0 points calculated in program, (lines 305-315)

X1, Y1 points input with light-pen (line 130)

Figure 5

Listing of Light-Pen Crayon (with Ellipse)

```
10  PRINT"LIGHT-PEN CRAYON—DRAWS ON 'FINGER DOWN'"
20  PRINT"KEY IN BACKGROUND COLOR"; :PLOT30:INPUT D$:PLOT12
30  REM-----DRAW BORDER-----
40  PLOT30:PLOT16:PLOT29:PLOT23
50  PLOT2:PLOT0:PLOT0:PLOT242:PLOT159:PLOT0:PLOT159:PLOT191
60  PLOT0:PLOT191:PLOT0:PLOT0:PLOT255
70  F=0
80  REM-----IS FINGER UP?-----
90  A=INP(251) AND 192
100  IF A=192 THEN 90
110  REM-----FINGER IS NOW DOWN, GET PEN XY-----
120  FOR Q=1 TO 100:NEXT Q
130  X1=INP(252):Y1=INP(253)
150  IF X1>157 THEN X1=157
160  IF X1<2 THEN X1=2
170  IF Y1>187 THEN Y1=187
180  IF Y1<4 THEN Y1=4
190  XC=INT(X1/2):YC=INT(Y1/4)
200  Y1=191-Y1
210  REM-----DRAW VECTOR-----
220  IF F<>0 THEN 250
230  X0=X1:Y0=Y1
250  PLOT3:PLOTXC:PLOTYC
260  PLOT2:PLOTX0:PLOTY0
270  PLOT242:PLOTX1:PLOTY1:PLOT255
300  F=1
305  R=R+.1
310  X0=70*SIN(R)+80
315  Y0=86*COS(R)+96
400  GOTO 90
```

Explanation of the Program

Line Number *Explanation*

20 PLOT 30 means "the next plot code will determine a new background color." INPUT D$ is a "dummy": this statement just stops and lets you use the keyboard to specify the color without generating a syntax error in BASIC. PLOT 12 means "erase the page." What this actually does is fill the screen with blanks of the new background color.

40 Sets up a background and foreground color so that the border will always be white on black. PLOT 29 means "next plot code will be a foreground color." Once the program is running, the user can change the colors of vectors by pressing special color select buttons on the keyboard.

50-60 Border is drawn all around the edges using vector mode.

70 First time flag (see line 220).

90-100 Finds out if pen is activated or not: "let A = the contents of input port #251 logically ANDed with the number 192." Explanation: First you have to know that input port #251 looks like this when the finger is up,

FINGER UP MODE

and like this when the finger is down,

FINGER DOWN MODE

Note: x means either 0 or 1, that is, it's not important for our purposes at this point—these are "don't care" bits. Secondly, logical ANDing does the following:

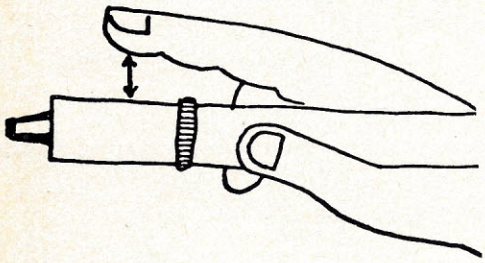


Figure 6

Touching or not touching the forward portion of the Compucolor pen can be programmed to activate it, that is, either the "finger up" or the "finger down" position can mean "draw." The motion of the finger need not be very large, as shown in Fig. 6. How this finger action is detected by a program will be made clearer by looking at the listing of our program. Here's the listing followed by a line-by-line explanation.

0 AND 0 = 0
0 AND 1 = 0
1 AND 0 = 0
1 AND 1 = 1
(no carries)

To examine just the first two bits, we do the AND (also called a "bit-wise AND") between the bits in port #251 and the pattern 11000000 (which is the binary equivalent of decimal 192).

```
    11xxxxxx = Port 251 when finger is up
AND 11000000 = 192 decimal
------------
    11000000 = 192 decimal

    01xxxxxx = Port 251 when finger is down
AND 11000000 = 192 decimal
------------
    01000000 = 192
```

The test in line 100 therefore means "if finger is up, keep looping back to 90, that is do *not* proceed with the program."

120 Means "do nothing 100 times." This empty loop allows time for data to transfer; otherwise "stale" values may be collected from the input ports holding X-Y values from the light pen.

130 Get X1 and Y1 from input ports #252 and #253 (pen).

150,160, 170,180 If pen is pointed outside the screen area it will send values outside the proper range (0-159, 0-191) causing wrap-around effects and other confusing errors. These IF statements "push" the values into a slightly smaller range (2-157, 4-187) so vectors will not be drawn over the border.

190 XC and YC are calculated to give a character "cursor" position corresponding to the light-pen coordinates X1 and Y1 (see line 250).

200 The Y coordinates from the light-pen are designated 0-191 starting at the *upper* left of the screen. Y coordinates for vectors must be designated 0-191 starting at the *lower* left. This calculation translates the Y values from the light-pen to proper plotting values for Y.

220,230 F = 0 only for the first vector to be drawn. At that point X0 and Y0 are not calculated yet, and they = 0. This would cause the first vector to always be drawn from the lower-left corner (coordinates 0,0) to the light-pen point. Line 230 causes the first vector to be drawn as a point at the light-pen X,Y.

250 PLOT 3 means "move the cursor to the following values." The next two plot codes must designate X and Y values suitable for characters; that is, 0-79 and 0-47.

260,270 Draw vector from X0,Y0 to X1, Y1.

305-315 Calculate new values for X0, Y0. In this version, X0 and Y0 define an ellipse. Any curve or line can be substituted to create variations. Note: Simply making X0 = X1 and Y0 =Y1 at this point will give a completely user-defined continuous line following the light pen.

400 Go back and check if pen is activated before drawing next vector.

This program contains an infinite loop. It can be easily halted on the Compucolor by pressing the break key, or terminated by pressing the line feed key. Examples of output produced by this program are shown in Fig. 7 and 8. You can see the outline of the ellipse generated by the program, and the modifications made by the light-pen. The ellipse-generating feature can be removed from the program as explained in the "Note" after the line 305-315 explanation. This makes the light-pen act as an electronic "paint brush." Fig. 9 shows an example of a bouquet of flowers drawn in this way.

The Compucolor is an ingenious machine, with features that make it one of the most fascinating personal computers developed to date. The BASIC is also very good (it has real string arrays, for example), and it's permanently stored in ROM. A user group was recently announced in this magazine, so new and better ideas on computing in "glorious color" will undoubtedly be coming along. Keep your eye peeled for the rainbow. ■

THE DIGITAL BRUSH

An Interview With *Star Wars* Animator Larry Cuba

First appeared in *Starlog* magazine, Number 12, March 1978.

By DAVID HUTCHISON

On the shifting sands of Tatooine nestles the small cottage of "Old Ben" Kenobi. Inside, Luke Skywalker and Ben listen to Princess Leia's plea for help via a holographic recording implanted in R2-D2. Also within the feisty 'droid's memory banks are the technical read-outs of the battle station *Death Star*. These plans may sway the balance of survival for Princess Leia's people in the fight against the Empire!

The man responsible for the physical creation of the little 'droid's memory readout is Larry Cuba. The sequence in the briefing room in which the schematic view of the *Death Star* appears on a huge electronic screen, displaying a simulated point of view of a pilot maneuvering straight down a trench on the surface of the *Death Star* to a two-meter wide thermal exhaust port, was accomplished by means of computer animation.

Computer animation is a process whereby the illusion of movement is bestowed upon inanimate objects by electronic means. In cel animtion, an artist must draw each frame of film by hand. Here the computer creates each frame which is then photographed and projected. (Or videotaped and televised.)

With *Star Wars* already in production, George Lucas issued a call for bids from companies and individuals to produce various bits of instrumentation animation—in particular the briefing room sequence. A number of computer artists and cel animators responded.

Some of the computer people had very sophisticated equipment capable of producing colored and shaded planes and forms. One computer artist even wanted to do most of the model sequences entirely on computers. George spoke with each of the artists and viewed their work, but Larry seemed to understand the kind of look that George wanted for the film.

When Larry was assigned the computer realization of the *Death Star* plans for the briefing room scene, he was asked to have the sequence photographed on 35mm film so the plans could be rear-projected during the filming of the briefing room scene with the rebel pilots. At UICC Larry would be using the Vector General 3D3I display and a PDP 1145 minicomputer. The sequence would be filmed off of the Vector General screen with a standard Mitchell 35mm camera rigged with an animation motor. The only thing lacking was the trench. John Dykstra's crew had not yet gotten around to building it.

John Dykstra and his team of modelmakers at Industrial Light & Magic (ILM) had begun to assemble the basic modular molds from which they would construct the model of the trench. The basic molds were constructed about two feet square in six different types. From these molds hundreds of casts were made in polyurethane foam. These modular sections were then cut up and assembled in a variety of basically random configurations to establish the sides and bottom of the trench as well as part of the *Death Star*'s surface area.

Larry took samples of each of the six to Chicago to construct his own computer trench. "There was no reason to have the computer sequence match the actual model precisely, since the audience would perceive the trench more in terms of a texture rather than an absolute configuraton," Larry explains. "ILM was chopping up the modular pieces to assemble the trench, so I did the same thing—building up the trench in the computer memory just like they were doing with the real thing.

"I photographed the six modules and traced them onto the Vector General data tablet with its electronic pen. By pressing the pen to the various points on the photographs,the modules were digitized—their x and y components entered into the computer." (The x component refers to the horizontal axis and the y to the vertical axis.) The z coordinate was entered manually.

The z coordinate (depth) was limited to about four or five different levels, so when entering the x and y components on the electronic tablet, Larry punched one of five buttons that he had programmed to represent the z coordinate at various levels.

"Then a program was written so that I could call up (from the computer's memory) the raw sections and combine them into the trench." The computer trench consisted of about fifty U-shaped sections (the two sides and bottom of the trench make a U). Larry called up sections of the modules, stretched or moved them around to build up the trench bit by bit."The trench information was stored away and another program written that would call up the sections sequentially, in the perspective of a pilot flying down the trench, and cue the camera to photograph a frame. I managed to get about thirty frames an hour into the camera once the program was running smoothly."

On the screen the *Star Wars* audience sees the computer realization of the trench sequence in the form of a "wire-cage" model rather than as a series of solid forms and planes. One of the early problems in computer graphics was the wire cage versus solid form display. At first computer programs could only call up figures in wire cage format. It was only a few years ago that programs were devised to remove the "hidden lines;" the program had to determine which lines would be "hidden" by a front surface or plane and remove those lines.

"When George Lucas specified the kind of animation he wanted for the scene, he knew enough about computer animation to ask for a true perspective without the 'hidden lines' removed. He wanted the trench and the *Death Star* to appear as wire cage figures with all lines and vertices visible. George thought that this sort of image would suggest 'computer animation' by having a very mechanical look."

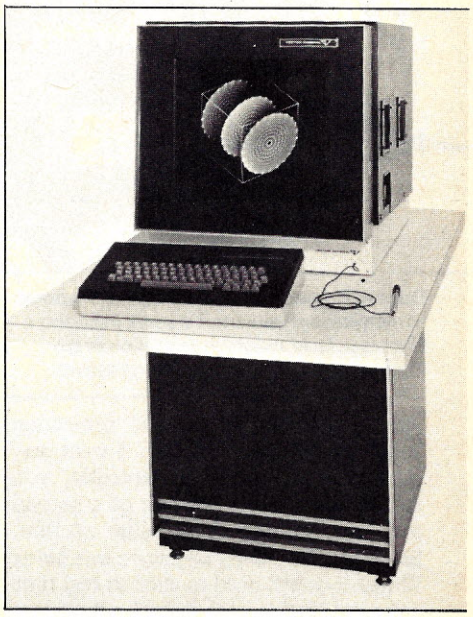


photo: courtesy Vector General

The Vector General Series 3's capabilities range from a simple two-dimensional graphics display to complex 3D transformations including scale, rotation about all three axes and variable intensity for depth cuing.

Science fiction as a genre often projects into the world of future technology. Larry Cuba suggests that in the future computers will be able to generate pictures of such quality that they will look as though they had been photographed by a camera. In the case of *Star Wars*, it was thought that such photographic realism might be confusing to the audience, so a wire cage model was specified so that the audience would readily understand that the images were to have been created by a machine.

Scene 135 from *Star Wars* with the rebel starpilots and navigators viewing the computer readout of the *Death Star* plans. The success of the rebellion depends on some small weakness in the *Death Star*'s design.

From start to finish, the entire sequence lasts only about 40 seconds on the screen. It took Larry and his two assistants T.J. O'Donnell and Tom Chomicz about two months to supply two minutes of animation.

The enormous number of points and lines on the wire cage figures that make up the representation of trench seem to flow with almost simultaneous precision. The computer doesn't handle all of these points simultaneously, but rather sequentially. It happens very fast, certainly, and it can appear to the eye to be happening all at the same time, which would be the case while observing a real-time system. A real-time system means that the computer is drawing successive frames as fast as thirty-per-second, which is what is needed to see the thing move smoothly on a TV screen. "There is a limit to how many of those points a computer can draw in a thirtieth of a second and in the case of the *Star Wars* animation with its true perspective image as opposed to parallel projection (one without depth cuing), I went way beyond that limit. Consequently, you take longer than a thirtieth of a second to put an image on a frame of film. Since the *Star Wars* sequence was being filmed it didn't need to exist in real time anyway. In this case it took about two minutes to complete each frame."

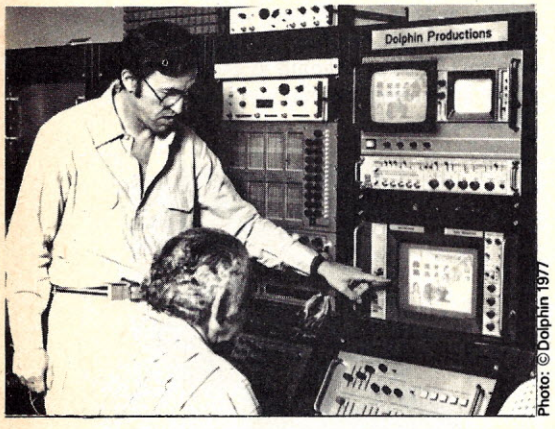


Photo: © Dolphin 1977

Dolphin Productions in New York is the home of many award-winning TV spots including a first place Gold Award in animation at the International Film and Television Festival.

There are, of course, displays more sophisticated than the Vector General, that could have computed the perspective more readily and probably done the flight down the trench in real time; the perspective transformation would be wired into the hardware itself, rather than generated by a separate program.

There are systems today that can generate shaded color planes in real time. One such system was developed by General Electric and built at a cost of $2,000,000 to train astronauts to land on the Moon. Similar systems are used to train airline pilots to land under a variety of emergency conditions.

Basically, Larry's system consisted of a $50,000 Vector General 3D3I graphics terminal with its dials and electronic data tablet, a $30,000 PDP 1145 minicomputer and standard alpha-numeric keyboard. "I set up a Mitchell 35mm camera with an animation motor in front of the screen and connected it to the computer so that a signal from the program could trigger the animation motor when the image was complete.

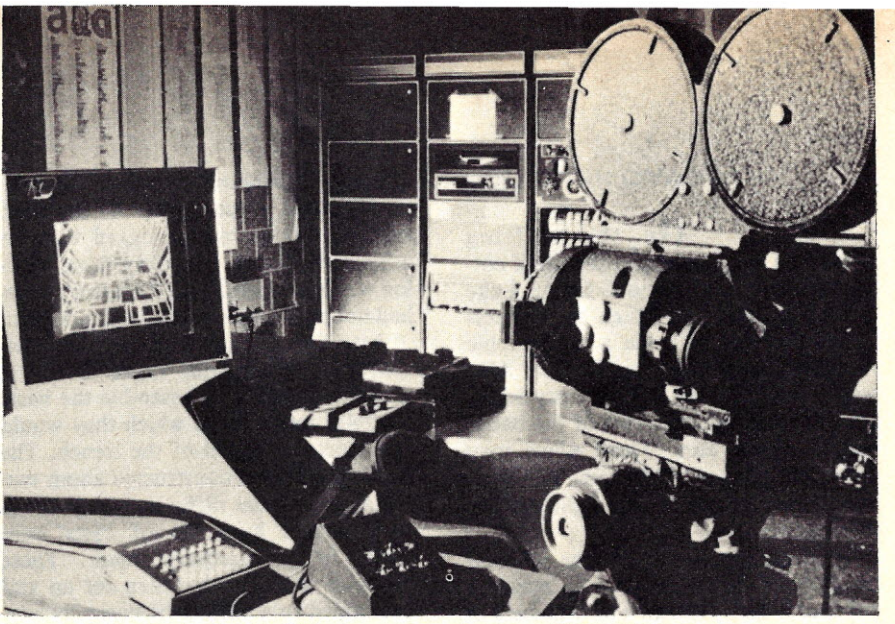


photo: courtesy Larry Cuba

Larry Cuba's setup for the *Star Wars* computer animation, with the PDP 1145 racked in the rear, the Vector General screen and Mitchell camera.

"The full length of the trench consisted of about fifty of these U-shaped sections. Well, you couldn't bring all fifty of these sections up on the screen at the same time. The computer brought up five sections at a time and it would take about 24 frames (one second) to go through one U-shaped section of the trench.

"So it was this continual shuffle of sections; never having more than five on at any one time. Now, of course, this means that ones at the back just sort of pop on. I had hoped to be able to just fade them in, bit by bit, by manipulating the intensity control to make them appear more slowly. But there wasn't enough time.

"The entire sequence was shot once, and that was it. Early on, I had a deadline of June first, but in early April the deadline was moved up to May fifth —lopping off three weeks. I had anticipated another six. I suggested that they wait and shoot the sequence in England blue screen; they could print the computer effects in later and have the thing perfect. But no, they wanted to rear project it so that the guys in the briefing room would play to the images while they were talking. Well, my first take worked. There were a couple of problems, but they edited around them."

The briefing room sequence is the only scene in *Star Wars* in which digital computer animation was used—other than for occasional background displays as part of the *Death Star* set. The effect was programmed in Tom Defanti's GRASS language. GRASS (GRAphics Symbiosis System) was written by Tom as part of his doctoral thesis for Ohio State. "It takes advantage of all the things that the Vector General does. The Vector General has a lot of image transformation hardware built into it, which allows you to do a lot of things in real time (with no processing delay). The language is designed for non-computer people. GRASS consists of very simple, straight forward commands which allow the students to work with the Vector General 3DI directly and manipulate the image by means of various dials and buttons.

"GRASS as a language makes it super easy for an educator or student to come in and call up a stored image (a crystal, molecule, etc.) and by means of the language manipulate the image, say rotation by a single dial, programmed in GRASS.

Suppose it is necessary to look at a particular molecule, a simple sugar for example, which has been named SUGAR. The molecule must be called up from the memory disk, shown on the screen, made larger or smaller and rotated for study. The commands would be typed out on the alpha-numeric keyboard in GRASS:

```
GET DISK SUGAR
SCALE SUGAR, DO
ROTATE SUGAR, X, D1
```

By means of these three commands the required molecule appears on the screen, its size can be changed by turning dial number "O," and it can be rotated around the x-axis (horizontal) by means of dial number "1."

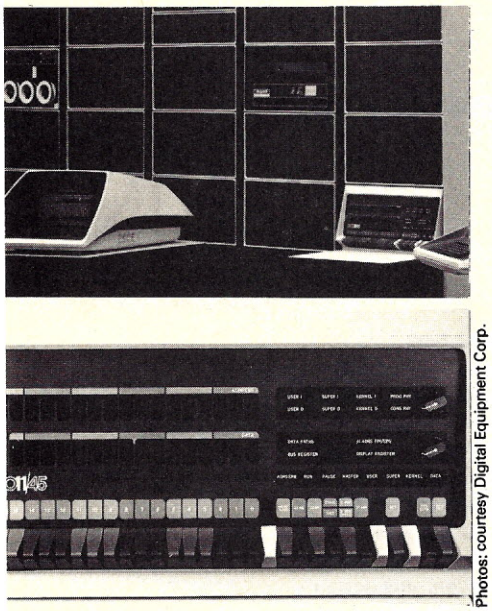
Sounds easy? It is. And what fun it must be to sit there and play with shapes and movement!

"The display can then be handled by an image processor—colored, mixed and recorded on standard videotape, 3/4 inch cassette or what have you." The

system has produced tapes in chemistry, mathematics, medicine and computer programs.

Additionally, since the system operates in real time, it has been used in performance in a live concert. Various monitors were spotted around the concert hall and one large Advent Video projector rigged. There are three performers. One performer programs the computer and operates the dials of the Vector General, creating the original image. The second manipulates the image processor and colorizes the image and the third performer creates music on an audio synthesizer to complete the video picture. A number of tapes have been made of these concerts and are generally available. PBS has broadcast a number of them.

But is it art? Mr. Cuba maintains that the computer and its peripherals are *tools*, like brushes and pigments to a painter. That the manipulation of these tools is by the mind of man and just as selectively controlled as any other fine art. "The computer as a tool gives us a new way to explore motion, movement and the kind of imagery that we have never really had the power to explore."



Photos: courtesy Digital Equipment Corp.

Digital's PDP 1145 introduced in 1972 as a large "minicomputer," has an internal memory of 262,144 characters and can handle 3.3 million mathematical calculations per second. The programmer's panel is shown in close-up.

Will we see more computer animation in motion picture making? So far it has had a very limited use. There was a sequence in *UFO: Target Earth* and *Futureworld*. All of the visuals aboard the ships in *2001* were cel animation masquerading as computer graphics. There were some in *Demon Seed*—one of the background display monitors ran a computer-generated model of an earthquake.

Ultimately, there is the possibility that the technology of producing curved surfaces, details, colored and shaded planes will replace some of the rather complicated special effects that can be created only by photography and optical effects.

Already computer controlled cameras could usher in the era of setless cinematography, in which the actors will work on giant blue-screen sets with all of the details added by computer (see *Magicam* in STARLOG # 9).

Computer video technology has found its way into commercial television. Numerous commercials and logos have made use of sophisticated video synthesizers to create, without the photographic camera or lengthy cel animation, the images required.

In New York City, Dolphin Productions uses the Scanimate video synthesizer to produce a good many of Madison Avenue's television commercials.

There are only five such machines in the world—originally built by Computer Image Corp. in Denver. The essence of the machine is that you can put down any picture or image and move it, transform it, distort it, flip it, color it right in front of your eyes and record the result on video tape.

The images can be saved, mixed or composited with other images and backgrounds so that little by little a completed sequence can be built up. Much of the credit must go to the enormous advances in recent years of computer controlled video tape editing. With the Scanimate equipment and the IVC 9000 video editing equipment a complete thirty second commercial may be produced in eight hours. The going rate, however, is $8,000 a day and up.

The process starts with an image, either a Kodalith on a light box scanned by a TV camera or a TV studio camera image. The image is then transformed in some manner, for example, compressed into a ball, colored and positioned on the screen.

Then the image can be moved and rolled in any manner around the screen. The Scanimate is operated by patching the video signal through various transforming modules in much the manner as an audio synthesizer. The movements are watched and tested at various settings until the client sees what he likes. Then it is recorded. Eventually a foreground and background reel is generated. At the end of the day the reels are composited, a sound track laid in and the client goes way with a complete TV spot tucked under his arm.

The advantage of the system is that the client can immediately see what he is getting without waiting for various laboratories and optical houses to process film and create effects.

Dolphin's use of the Scanimate equipment allows them to have almost any job out in two days at half the cost of the average commercial. Certainly if the effects of figures twisting, stretching, zooming, strobing, or squeezing against a "three-dimensional" background were attempted with cel animation, the cost would be prohibitive.

The Scanimate, however, isn't intended to compete with cel animation, but to produce visually effective animation on the spot, with the client watching.

Certainly the potentials of computer animation have only been suggested. Much is still unrealized, waiting for the man with the ideas and visions to use these new tools. ■

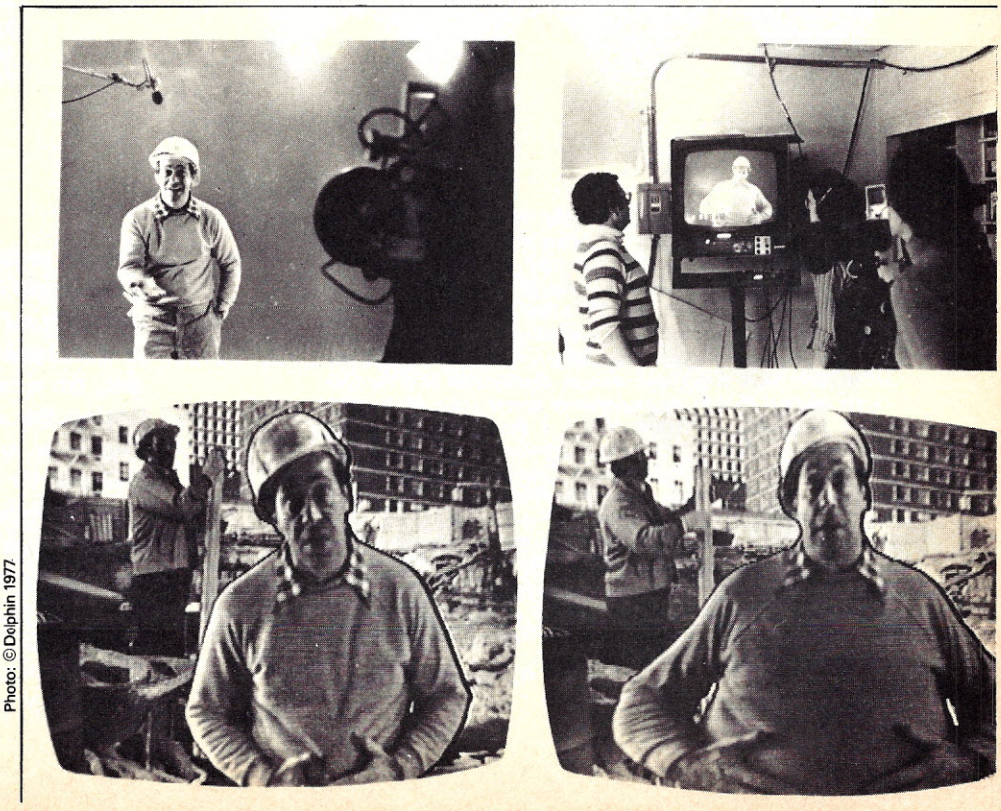


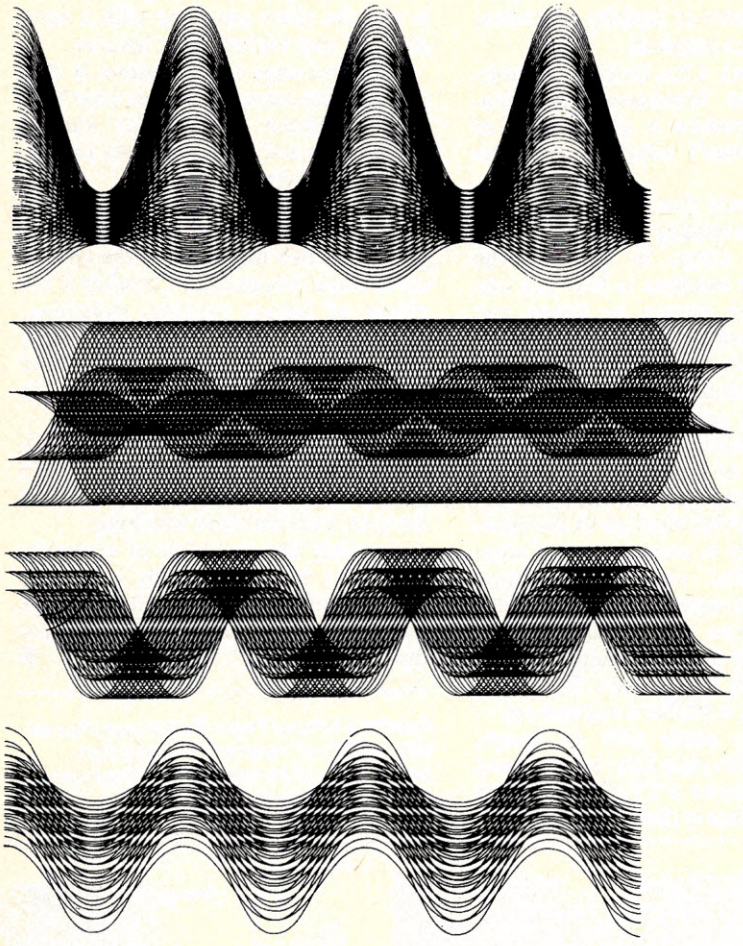
Photo: © Dolphin 1977

Dolphin's famous Pepto-Bismol spot dramatically inflates a "hard-hat" afflicted with indigestion—all by means of the Scanimate electronic video synthesizer.

With dazzling speed, the computer has reexplored a succession of blind alleys in the visual arts. But the future looks different. "Whatever the technical route, we are on the verge of realizing an entirely new artistic mode."

Idols of Computer Art

ROBERT E. MUELLER



These pure sine waves of differing amplitudes by Bruno Sonderegger are Lissajous Variations in which step voltages and frequency changes are used. Since the sinusoid is a "natural" function, this design falls into the Idol of Nature category.

The computer is dangerously close to being our modern version of the kaleidoscope. The twists and turns of programs give unexpected variations of form that seem to be strikingly beautiful. But is it art? What is beauty? Many people in the computer field do not seem to realize that there is a long history of aesthetic investigation into this problem. I am an artist (with woodcuts in many museums including the Museum of Modern Art in New York City), and also an aesthetician who bridges science and art (I have an engineering degree from MIT and a degree in philosophy from NYU). I have given computer graphics a lot of thought since its inception—see my book The Science of Art: The Cybernetics of Creative Communication, *chapter 8, The Computer Apprentics (Day, 1967.) This article entitled IDOLS OF COMPUTER ART, reprinted with permission, was published originally in* Art In America *(May-June 1972.) I thought it might be of interest to the readers of* Creative Computing. *It summarizes the pitfalls and limitations of computer graphics as fine art, for the benefit of people who either take themselves too seriously, or who would like to try and take themselves more seriously as artists.*

— Robert E. Mueller

It is not surprising that a device as powerful as the computer should influence art—the latest in the long line of technological developments to do so. While I believe it will ultimately cause a minor revolution in all of the arts, the results to date are exceedingly poor and uninspiring. But all new media take some time to be assimilated—not to mention the economics of making them available for something so nonutilitarian as the arts.

Since Pythagoras, music has of course been far more tractable than visual art to mathematical, and thus eventually computer, manipulation. Johann Joseph Fux set the stage for classical music in 1715 with his *Gradus ad Parnassum*, the basic treatise to codify counterpoint in music. A similar mathematical impulse prompted Helmholtz to write his *Sensations of Tone* in 1863, and also

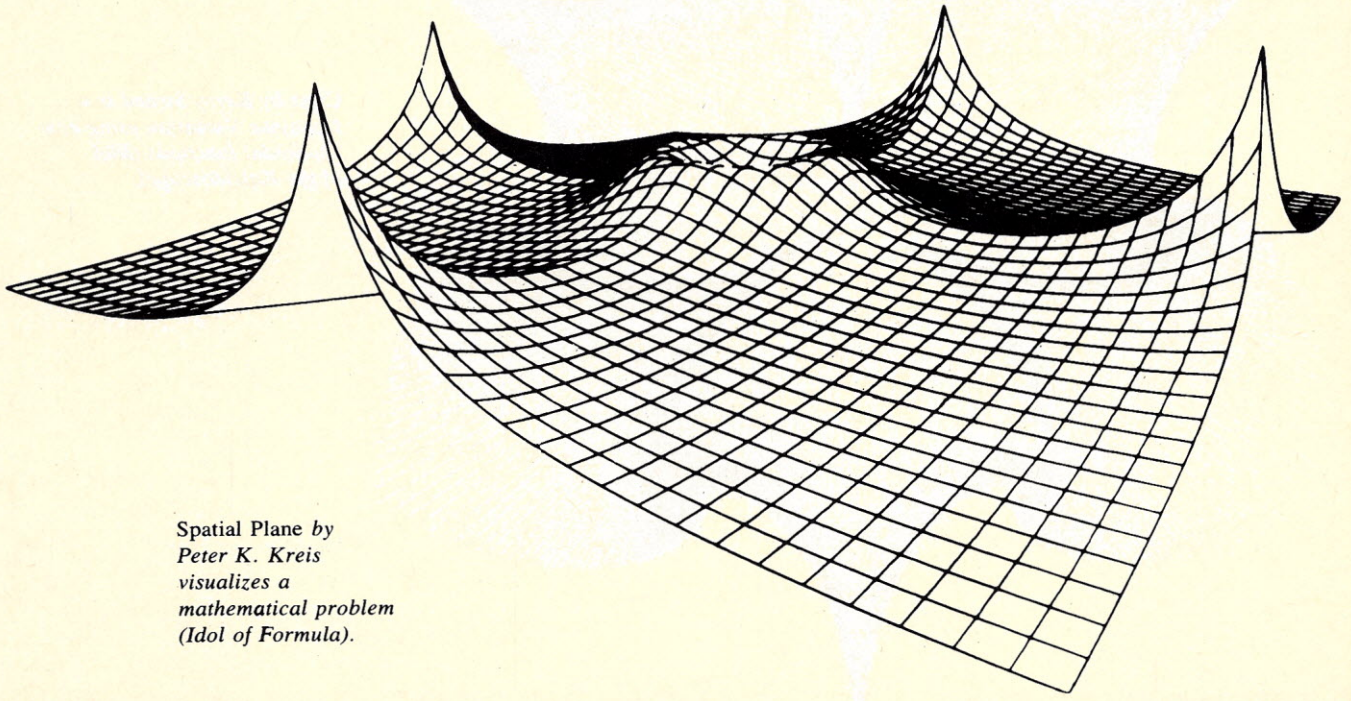
Paul Hindemith his *Craft of Musical Composition* in 1936, both updatings in disguise of Pythagoras' drive toward ordering musical notes. Schönberg took a different tack when he introduced the arbitrary, acoustically independent technique of the twelve-tone row; it represents the triumph of the urge toward mathematical abstraction over empirical necessity, the same urge Euclid demonstrated when he lifted geometry out of the practical world and put it on the plane of pure thought.

The computer permits this very old desire to organize tones or create new ones to be accomplished with great ease, and at a level of organization far beyond our capacity for perceptual discrimination. Milton Babbitt has pushed the impulse to mathematize musical quantities and qualities to its limit, subjecting harmonic, timbric, rhythmic and dynamic variations to the dominance of a single mathematical logic—a feat possible only with a computer. And specialized computer languages are in existence to increase the spectrum of possible tones, all generated directly on magnetic tape with little technical knowledge of the computer required to use them.

One would expect that mathematical ideas would similarly influence the manipulation of light and color. Although color organs are very ancient (Aristotle refers to the relationship between color and music in his *On Sense),* no artist has managed to apply mathematical virtuosity to visual phenomena for expressive purposes. Indeed—with a few notable exceptions—artists have remained somewhat aloof from the technological know-how our age has contributed toward color theory and production. But with the recently invented devices for creating or handling color (e.g., color phosphor cathode-ray tubes, electroluminescent screens, or holographs) this might change. And given the computer to control them, new opportunities for inventive manipulation will no doubt open up, limited only by the availability of such media and their comprehension by interested artists.

While these technological breakthroughs are being ironed out and made available, the computer specialist has been engaged in a private, often playful investigation of the computer's potential for making graphic curiosities. These productions are related to the fantastic curves invented by nineteenth-century mathematicians, and before that to constructed geometrical shapes the Greeks derived from conic surfaces. Of course the computer specialist doesn't realize it, but his computer graphics are exactly like those unpredictable and originally meaningless curves that just happened when geometric elements were fiddled with indiscriminately. Mathematicians assigned them highly romantic names: Devil's curves, Rose curves, Witches of Agnesi, Syntratrixes, Curves of Pursuit, Loxodromes, Caustics. This activity preceded the invention of analytic geometry, and was perhaps instrumental in its birth. Computer graphics may be a similar paradigm of some future computer mathematics.

Judging by the results, three major classes of computer graphics are being produced. The first, which I call "Lissajous Variations," has its counterpart in the traceries of pendulums and their mechanical or elec-



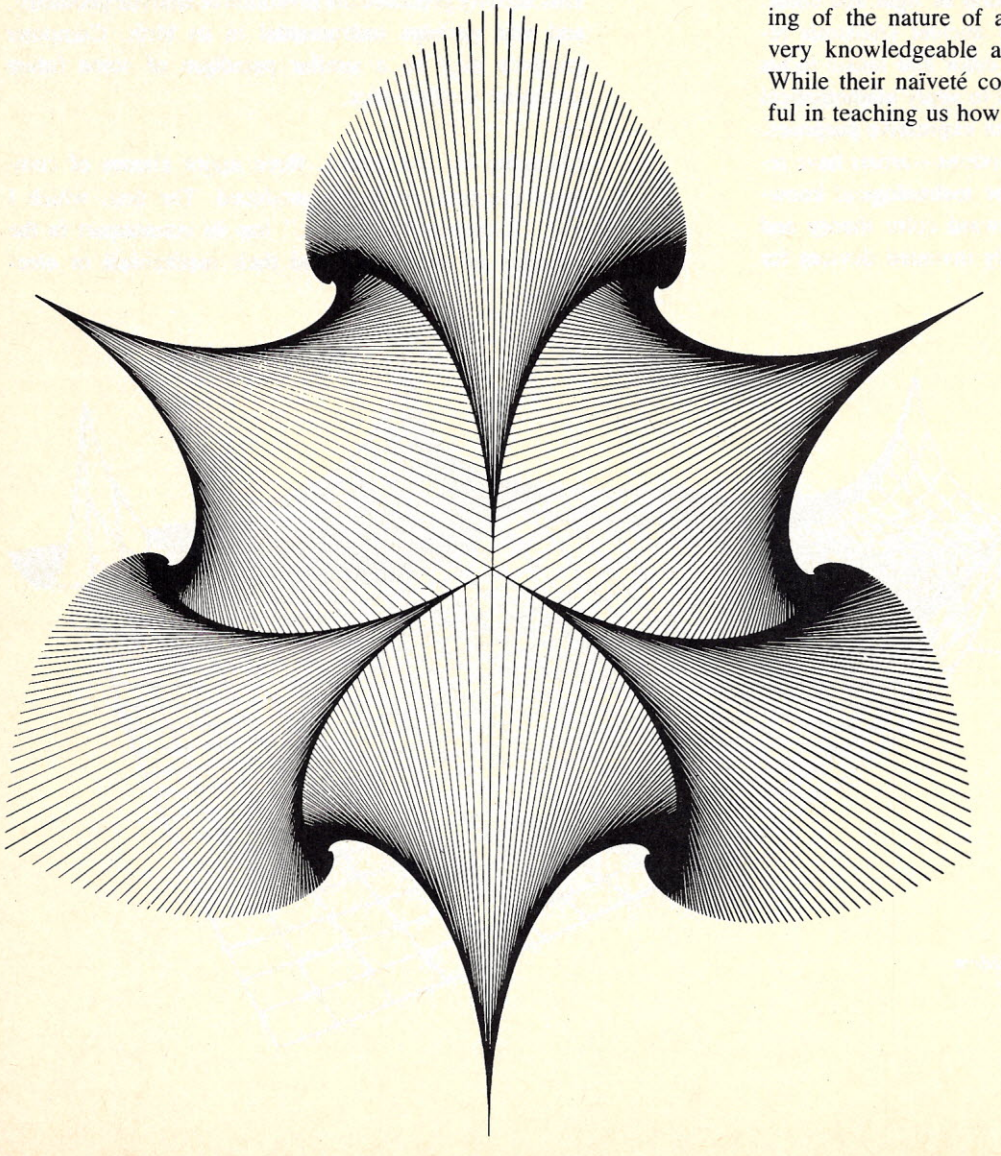
Spatial Plane *by Peter K. Kreis visualizes a mathematical problem (Idol of Formula).*

tronic equivalents. They were first investigated in 1815 by Nathaniel Bowditch (and were sometimes called Bowditch Curves), but studied in detail by the French mathematician Lissajous whose name they now bear. Lissajous figures are generated when two or more vibrating systems act on one another. These systems can be mechanical, like a pendulum supported by two or more legs; or they can be electronic, as when signals interact and are viewed on an oscilloscope. When the vibrating systems are generated by a computer and made visible by a pen plotter or cathode-ray-tube peripheral device, the resulting Lissajous Variations can produce complex and highly interesting forms.

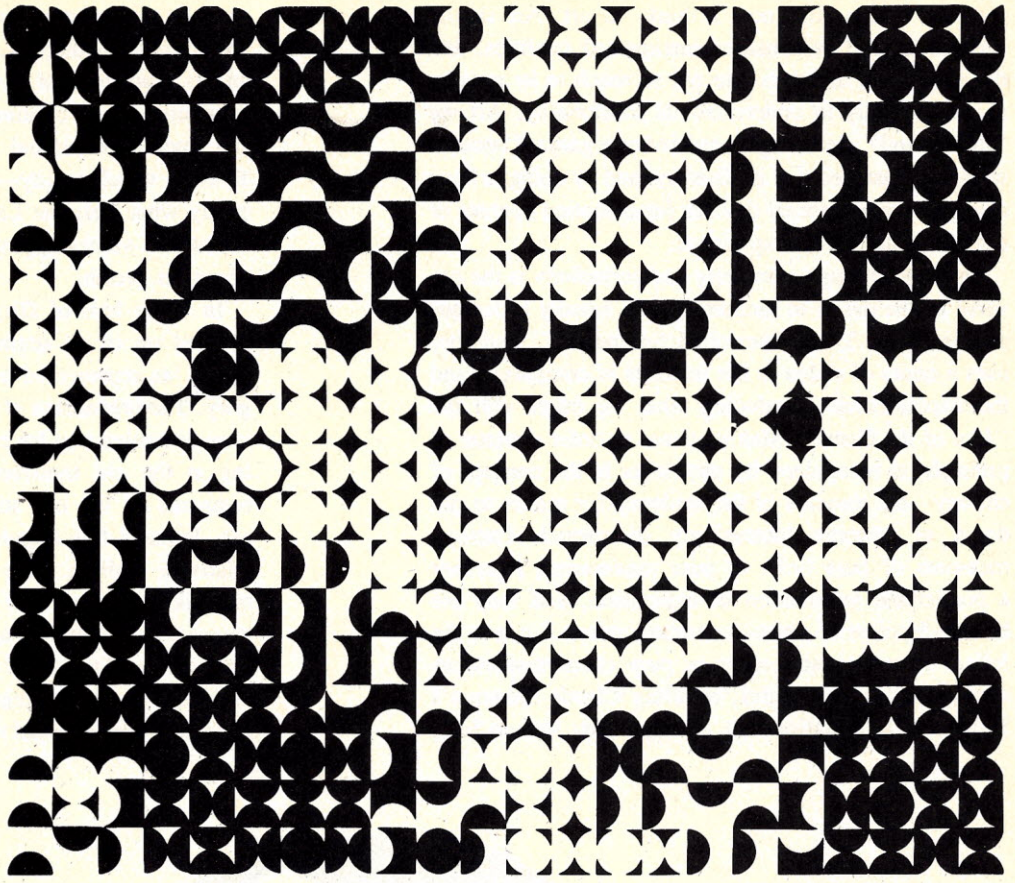
The second class of computer art, "Transformations," takes some recognizable picture or curve or function, and subjects it to a consistent alteration. It is related to the distortions of fun-house mirrors, and also to the technique for reducing photographs to halftones for the purpose of printing. But instead of converting the picture to a series of dots, this technique breaks down the original into many different elements, sometimes lines, spirals, wavy variations or perhaps arbitrary shapes or symbols.

The third and most interesting class of computer graphics, which I call "Controlled Serendipity," has the most artistic potential. This technique uses a given visual shape or form, either one derived from a photograph or picture or from a mathematical curve—or even from a new form made directly by the programmer—and subjects it to various random manipulations. The resulting pattern is observed and alterations are made on the original shape in order to see what happens the second time around. This is a feedback process in which randomly discovered elements can be emphasized or attenuated at will by the operator. It imitates a mode many artists use—the incorporation of accident—except that the randomness is introduced on purpose, in most cases through random-number generators. In science this introduction of randomness is sometimes called "dither," and B. F. Skinner has called Impressionism "realism with dither."

But computer art to date suffers from basic limitations due, in large measure, to a lack of understanding of the nature of art. Computer specialists are not very knowledgeable about the history of abstract art. While their naïveté could be refreshing and even helpful in teaching us how to exploit a totally new medium,



Crest *by Kerry Strand is a Lissajous Variation using non-sinusoidal functions (Idol of the Kaleidoscope).*



Zdeněk Sýkora based this Controlled Serendipity graphic on a computer program which distributed the design elements over a grid, producing playful variations on simple shapes (Idol of Game).

it has prevented them from achieving anything but the most superficial designs. They do not realize that Duchamp and Gabo, for instance, experimented with similar mechanically and photographically originated graphic ideas in the early decades of the century.

The errors into which specialists fall when attempting to apply the computer to art, one may call the Idols of Computer Art, in the same sense that Francis Bacon's Idols of Science stood for the traps of scientific theorizing. I feel that, to date, computer artists have been preoccupied with six intimately related false notions, which may be called "Idols of Nature," "Idols of the Formula," "Idols of the Kaleidoscope," "Idols of the Game," "Idols of Disguise" and "Idols of the Eye." Since these Idols apply to fundamental ways in which perceptual material is organized, it is reasonable to suggest that they also apply to music and any other art form subjected to computer manipulation.

All three categories of computer art can be used to generate what I call the "Idol of Nature," or that tendency to use natural order as a basis of patterned form. Things in nature such as crystals or flowers, the human body, landscapes, and so on, can become a meaningful part of a work of art. But when nature is simply reflected—increasingly possible as computerized techniques advance—its value as art becomes problematical. The results may be impressive, but they lack the necessary human insight and intervention, remaining "art-like" rather than becoming art. For example, natural forces are being released or channeled when Lissajous patterns are formed. These patterns represent not art so much as a methodological realization of forms implicit in nature, even though that nature is, of course, quite removed from a flower or sunset or crystal.

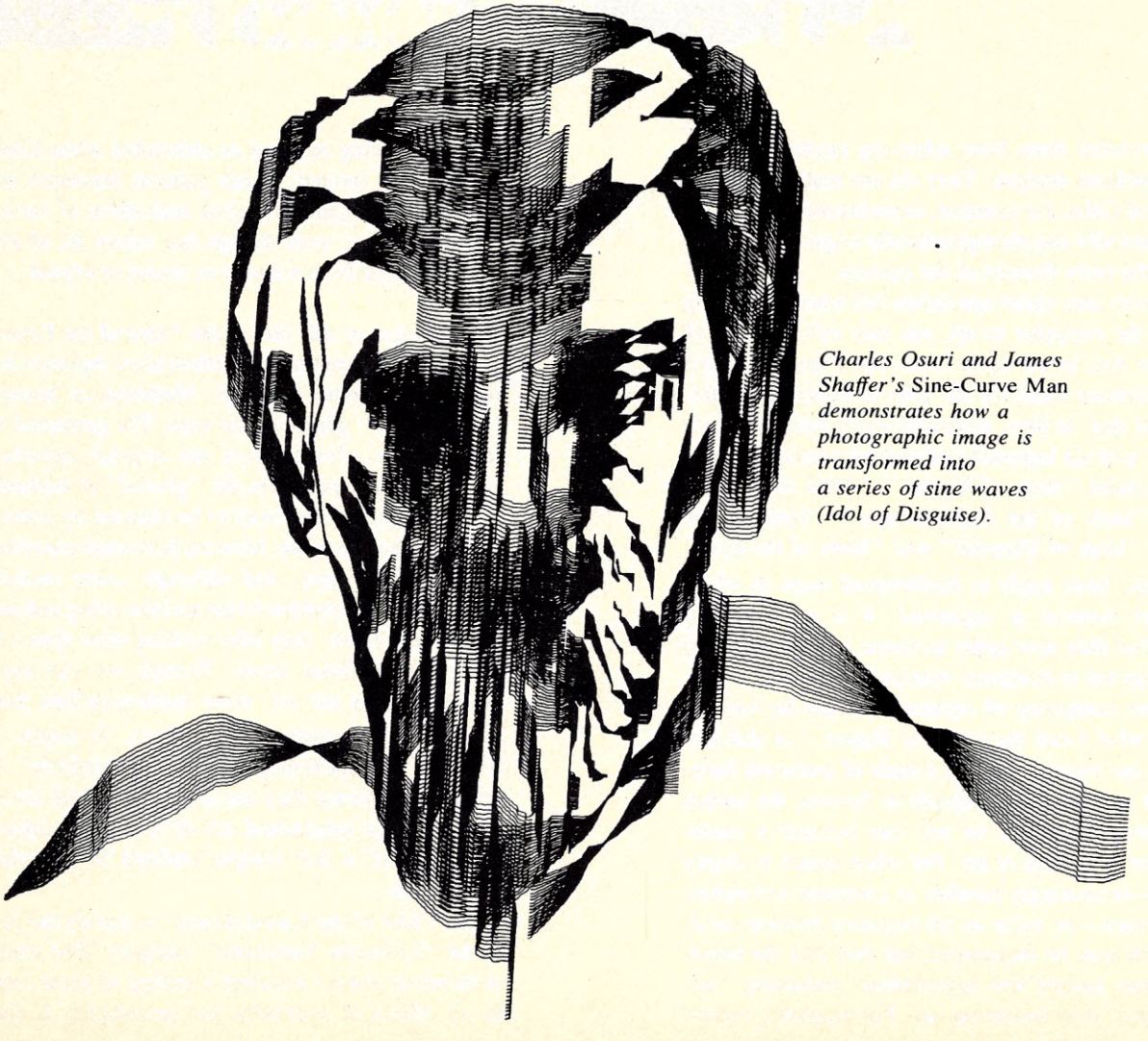
A subclass of this Idol is the "Idol of the Formula," in which a predetermined mathematical equation is used to *generate* some structure. Mathematics becomes a "new nature" generated by man. The generated forms may not be obvious from the original formula, but since they are implicit in the "givens" of mathematics or programming, they wait to be released by some technique for the eye to see. Most mathematical constructions fall into this class, and although conic sections or topological soap-bubble forms are interesting and perhaps highly suggestive, they offer nothing other than a rather empty inspirational force. Though we can say that mathematics is not art, some mathematicians think of themselves as artists of pure form. It seems clear, however, that their elegant and near-"esthetic" forms fail as art, because they are secondary visual ideas, the *product* of an intellectual set of restraints, rather than the *cause* of a felt insight realized in and through visual form.

The "Idol of the Kaleidoscope" is mainly the product of the "Lissajous Variations" category. The mirroring of elements always transfers a feeling of great order, as do all effects of symmetry and periodicity. It leads to

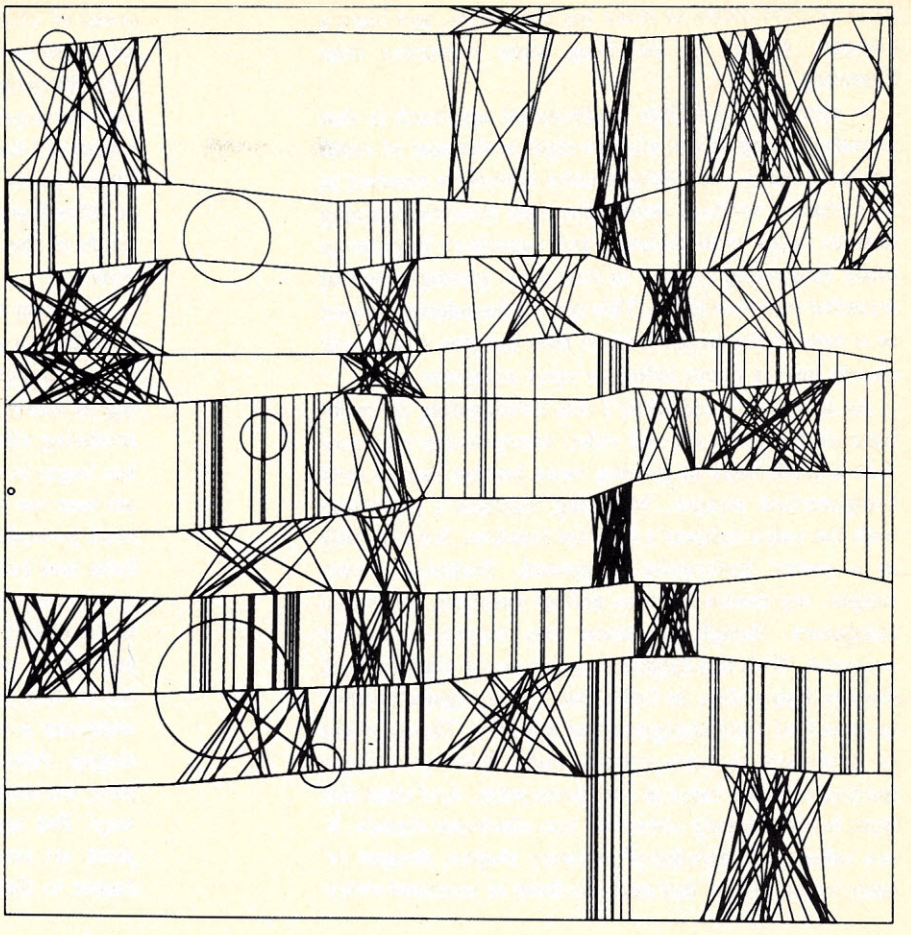
pleasant design, but not, I think, to art. Art should surprise us and demonstrate unexpected qualities, and the surprise or shock is due not so much to its novelty as our inability to understand its irrationality. Art we "understand" seems highly ordered or organized because we have exhausted it of its disarray, and by doing so have changed our perceptual devices for detecting degrees of disorganization in our experience.

Ordering through symmetry and periodicity is the obverse of the desire to randomize—an equally fallacious end in itself. The "Controlled Serendipity" category uses a quota of chaos in the interests of complexity and the unpredictable, producing what we might call the "Idol of the Game." Making chaos or order a matter of principle is recognized by physicists in the concept of entropy—the measure of the tendency for matter to run down or become increasingly disordered. Entropy measures of pure order or pure randomness represent a predictable termination of expression, and they are both null-points of artistic communication. But it is the *failure* to attain pure order or pure chaos that makes such attempts meaningful, recalling Claude Shannon's Theory of Information that all significant human form must lie somewhere between them. Fortunately between these extremes there is, as Rudolph Arnheim points out in his *Entropy and Art*, an incredibly rich variety of structures. Their continual evolution enables us to order our mental-perceptual mechanisms into conventions through which reality—and art—is interpreted. This is partly what Suzanne Langer means when she says that art attains values appropriate to our intuitive judgment about its worth.

In "The Idol of Disguise" some form or design is dressed up in an attempt to give it a new perceptual status, as represented by the "Transformations" category. The prevalence of this type of alteration makes it a very deceptive trap for computer artists. We enjoy looking at the old transformed into the new, with some remnants of the old still present to tell us where we are. Perhaps this impulse prompted Picasso to include relatively realistic nude bodies along with the African masklike faces of *Les Demoiselles d'Avignon*. Total newness is incomprehensible—even if it were possible. The "Idol of Disguise" represents the repackaging urge manifest in art as eclecticism—one way to sell the novel to a conventional world.



Charles Osuri and James Shaffer's Sine-Curve Man *demonstrates how a photographic image is transformed into a series of sine waves (Idol of Disguise).*



Frieder Nake has translated a picture by Paul Klee to create a "new" work (Idol of Disguise).

The "Idol of the Eye" plays on that organ's perceptual capacities for novel effects, often to the point of saturation and sensory overload; many "Lissajous Variations" fail because of this. This "Idol" is illustrated by optical illusions, the visual "enigmas," which are the Op artists' "nature." We now begin to enter those gray areas where our perceptual apparatus causes subtle and important inflections. Music, for example, depends upon the nonlinear qualities of the ear to generate the hierarchies of harmonic importance. And of course the eye's physiological limits play an important role in our reaction to color harmonies. But we do not yet know exactly how optical illusions can be exploited most fully, although artists like Vasarely or Bridget Riley have begun to show us meaningful artistic applications.

Is it possible to imagine more viable computer art? The greatest single limitation on computer graphics seems to be the peripheral devices, the input and output equipment by which people can enter their visual ideas into the computer and receive them back. Another problem is that the artists' visual field of interest is far more complex than technicians realize. Consider line, for instance: the most superficial study of artists' drawings reveals nuances of stroke, pressure and texture inaccessible to the monolithic ball-point stylus or the cathode-ray beam, moved step by step across a sleek, homogeneous visual plane.

At this point in the development of computers, the visual ideas with which they deal are so simplified that they bore the sophisticated artistic mind to death. What is needed is an electronic medium offering as much control and variety as, say, watercolors. This is not inherently impossible, though most computer designers respond to such an opinion with a look of total incredulity—especially in the area of peripherals. The answer may lie in a television system linked into a computer, with some direct manual control provided for the artist. Such a system must allow human manipulation of as small or as large an area as the artist desires, and could theoretically be as subtle and precise as any classic artistic medium. Hands become the crux of human involvement with visual media, because without their virtuosity minds are stranded.

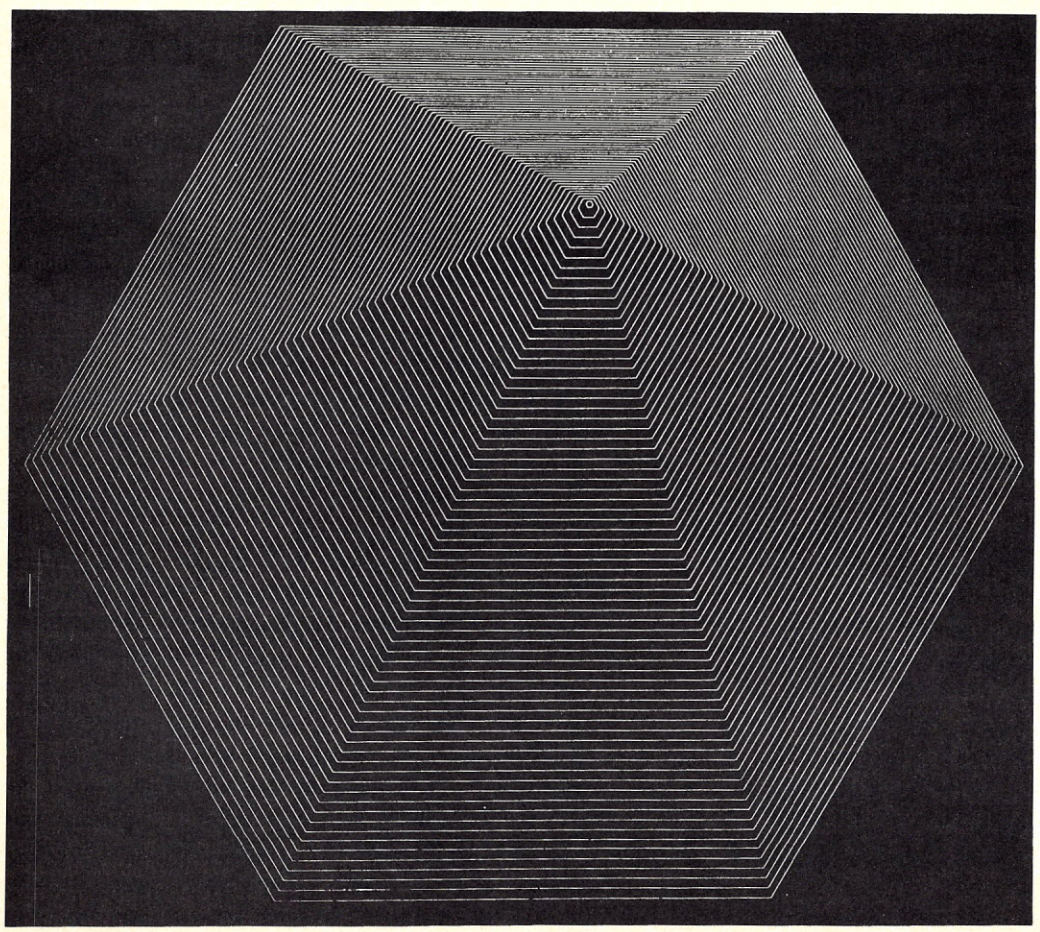
Although no existing electronic medium gives an artist direct manual contact with the computer's visual memory or computational powers, video control clearly lends itself to computer adaptation. The most interesting idea so far was conceived by Lee Harrison III. His device can be seen any evening on television, manipulating the titles and formats of commercials. Harrison's device splits into sections any given input image placed on a pickup screen. The operator can manipulate these sections one by one, varying their relative positions, distorting their shapes, sizes, colors, and so on; and images can be brought together or overlapped in full color for photographing or video taping. The images are controlled by analogue-computer circuits, but an artist

must twiddle knobs to make his alterations, and learn a complex system of switching more restrictive than liberating.

A more direct if less professional approach is that of many young artists who go right to the seat of video artistic control—attacking a color television receiver in its circuitry, working with video tape systems, learning how to fiddle with resistors and capacitors in order to make interesting images in real time, photographed or recorded on video tapes. This type of floundering around *in* a new medium can lead to new insights which will clearly have a direct influence upon computer art.

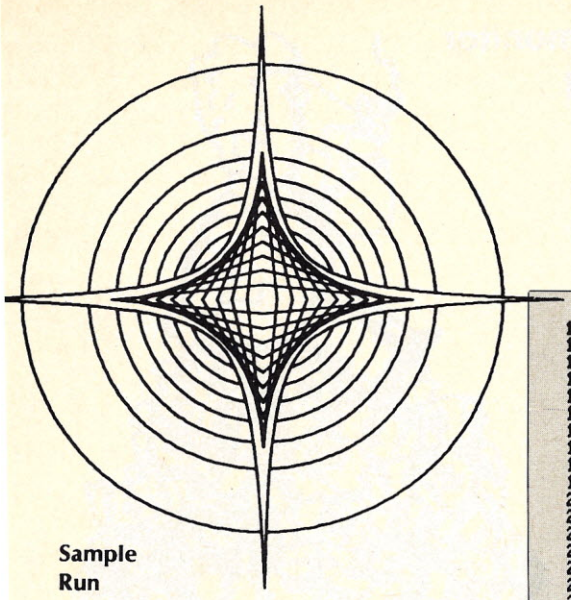
An idea conceived while I was investigating the problems of computer art and video manipulation may provide a crude start in gaining more precise control over computerized images. Marrying Harrison's perfection with the video tinkerer's urge for freedom, it exists only as a patent application at present. Technically very simple, my device requires neither analogue nor digital computers, though of course they would expand its potential. With this system a person could draw by hand directly into video, in full color, using regular brushes or pens (but without pigments of any sort). This provides an extremely delicate control, right down to a single hairline of video input at any given point. And since this input is immediately converted into electronic signals, it can release a repertoire of arbitrary shapes, designs or other visual effects that emanate from or surround every point of contact with the brush or stylus. Through a keyboard control, these other shapes can be "played" point-by-point by the operator's left hand while he draws individual points with his right. This in effect multiplies a person's hands, permitting him to draw circles, entire lines, bands of colors, or many different geometric or other forms anywhere on the screen simply with one touch of the brush to the surface of the input "draw" screen.

Whatever the technical route, we are on the verge of realizing an entirely new artistic mode. An electronic-video-computer visual medium is as different from painting as film is from theater. As more interesting ways of rendering visual form are developed, and as the specialists begin to understand the limitations of their device. I am sure we will begin to see much stronger results. The most powerful impact will be, I feel, on mathematical form and problems of pattern recognition, an area on which computer specialists are already at work. They will probably discover that computer graphic productions are not so much art as they are new insights into the forms that must be explored in order to make the computer a more useful tool for dissecting generalized shapes. Afterward, perhaps, with some luck and know-how, the artist can begin to use the computer in his own way. But computer graphics will never become computer art until the technical processes become second nature to their artist-manipulators. ■



Shift No. 2, 1969, *by Auro Lecci, is a design in which restraints are transformed to create an expanding septagon around a point (Idol of the Eye).*

Computer Art: Stargate



Sample Run

Joe Jacobson

I create computer art using either a CRT output or a mechanical plotter. The first step is to find an idea or theme. Then I figure out a computer routine that will generate this design. The last step is to write the program and run it. Some programs embody generalized geometric routines and will draw a range of different pictures in response to a variation in the parameters entered through the keyboard. Other programs are designed solely to draw a particular picture that is envisioned at the beginning.

The picture shown, "Stargate", was drawn with the latter type of program. The geometric ideas came from several sources. Kelly Freas, the well-known science fiction artist, had shown me a geometric design he created for a logo. Christian Kuebler, a fellow experimenter in computer art, had suggested an interesting geometric algorithm several years ago. And I had an idea I wanted to use sometime. It occurred to me to make a synthesis of these design elements in one picture, and "Stargate" is the result.

This picture was generated on a Tektronix 4051 terminal, which can be used as a stand-alone microcomputer. It uses BASIC and has a package of graphics routines, and provides the user with 8K of RAM. The picture is displayed on the terminal storage CRT screen, and the system includes a hardcopy machine.

I've collected about a hundred such "plotter art" pictures, done over several years by myself and a few friends. ■

```
100 REM "STAR GATE"
110 PAGE
120 REM
130 A=3.3176
140 B=0.3214
150 C=10
160 WINDOW -10,10,-10,10
170 VIEWPORT 15,115,0,100
180 MOVE 0,C
190 FOR X=0 TO C STEP 0.1
200 Y=A/(X+B)-B
210 DRAW X,Y
220 NEXT X
230 FOR Q=0 TO 30 STEP 5
240 I=8-SQR(Q)
250 MOVE 0,I
260 J=1
270 FOR T=0 TO PI/2 STEP PI/(25*I)
280 X=I*COS(T)
290 Y=A/(X+B)-B
300 Z=I*SIN(T)
310 IF Z<Y THEN 370
320 IF J>1 THEN 360
330 MOVE X,Z
340 J=J+1
350 GO TO 370
360 DRAW X,Z
370 NEXT T
380 NEXT Q
390 MOVE -C,0
400 FOR X=-C TO 0 STEP 0.1
410 Y=-A/(X-B)-B
420 DRAW X,Y
430 NEXT X
440 FOR Q=0 TO 30 STEP 5
450 I=8-SQR(Q)
460 MOVE -I,0
470 J=1
480 FOR T=PI/2 TO PI STEP PI/(25*I)
490 X=I*COS(T)
500 Y=-A/(X-B)-B
510 Z=I*SIN(T)
520 IF Z<Y THEN 580
530 IF J>1 THEN 570
540 MOVE X,Z
550 J=J+1
560 GO TO 580
570 DRAW X,Z
580 NEXT T
590 NEXT Q
600 MOVE -C,0
610 FOR X=-C TO 0 STEP 0.1
620 Y=A/(X-B)+B
630 DRAW X,Y
640 NEXT X
650 FOR Q=0 TO 30 STEP 5
660 I=8-SQR(Q)
670 MOVE -I,0
680 J=1
690 FOR T=PI TO 1.5*PI STEP PI/(25*I)
700 X=I*COS(T)
710 Y=A/(X-B)+B
720 Z=I*SIN(T)
730 IF Z>Y THEN 790
740 IF J>1 THEN 780
750 MOVE X,Z
760 J=J+1
770 GO TO 790
780 DRAW X,Z
790 NEXT T
800 NEXT Q
810 MOVE 0,-C
820 FOR X=0 TO C STEP 0.1
830 Y=-A/(X+B)+B
840 DRAW X,Y
850 NEXT X
860 FOR Q=0 TO 30 STEP 5
870 I=8-SQR(Q)
880 MOVE 0,-I
890 J=1
900 FOR T=1.5*PI TO 2*PI STEP PI/(25*I)
910 X=I*COS(T)
920 Y=-A/(X+B)+B
930 Z=I*SIN(T)
940 IF Z>Y THEN 1000
950 IF J>1 THEN 990
960 MOVE X,Z
970 J=J+1
980 GO TO 1000
990 DRAW X,Z
1000 NEXT T
1010 NEXT Q
1020 WINDOW -10,10,-10,10
1030 VIEWPORT 40,90,25,75
1040 N=10
1050 D=1
1060 M=1
1070 T=0
1080 R=M*D
1090 X=R*COS(T)
1100 Y=R*SIN(T)
1110 MOVE X,Y
1120 IF T>1.5*PI THEN 1200
1130 M=N+1-M
1140 T=T+PI/2
1150 R=M*D
1160 X=R*COS(T)
1170 Y=R*SIN(T)
1180 DRAW X,Y
1190 GO TO 1120
1200 M=M+1
1210 IF M>N THEN 1230
1220 GO TO 1070
1230 END
```

Author at Tektronix 4051, used as a stand-alone microcomputer to generate Stargate.

Joseph P. Jacobson, 18-C Franklin Drive, Maple Shade, NJ 08052

This article shows how a microcomputer can monitor an individual's actual financial behaviors and interpret them in terms of strategy systems.

BASIC Financial Behaviorism or First Steps Toward a Real Budget

Dennis J. McGuire

As microcomputers become increasingly available to the general population for home use, the practicality of sophisticated financial management for the consumer has arrived. A survey of available software products has revealed that serious programs for the strategic management of personal expenditures have not yet found their way to the marketplace. Catalogue descriptions and articles on the subject offer rather trivial programs such as the ability to compare the amount of money a family spent on food this month with the amount spent during the same month a year ago; and, of course, the abundance of games includes a few on theoretical budget strategies. This article shows how a microcomputer can monitor an individual's actual financial behaviors and interpret them in terms of strategy systems. Since certain data-processing developments are necessary for the strategic management of financial behaviors, both in practical software applications and hardware devices, these are discussed in terms of the response of the microcomputer industry to a recent survey of its interest in financial behaviorism, and of the consumer-research industry's interest in it.

An individual's actual financial behaviors must be monitored over a period of time if his proposed budget strategies are to merit any serious consideration. Each act of buying some goods or service is a decision that provides a bit of information about the individual but the meaning of each act can be ascertained only when it is viewed in the context of all the other buying decisions that have been made during the same time period. A computer makes it relatively easy to record expenditures by account number and can show what percentage of a month's spending was allocated to each account. Ranking the account expenditures by percentage reveals a budget strategy. This strategy is the one that the person is actually operating with, whether or not he is fully aware of it as a strategy, and must be the basis for considering possible alternatives that are necessary for the purchase of some desired product or service.

Many people are interested in monitoring their financial behaviors out of sheer curiosity about themselves and a simple desire to know exactly how they are spending their money. An additional motivation for such monitoring is indicated by a basic hypothesis of financial behaviorism; namely that an exact, comprehensive and completely factual accounting of one's financial behaviors will lead to the more efficient use of one's money to obtain the goods and services one really wants. Experiments conducted to verify this hypothesis have given indications that not only is this true but the individual learns to plan and carry out operations that require financial decisions such as relocating, buying a house, going into business, etc., much more efficiently than was the case prior to the practice of financial behaviorism.

Dennis J. McGuire, PhD, 4281 Henderson Place, Syracuse, NY 13219.

The SPEAS

In the analysis of financial behaviors only expenditures of disposable personal income are of interest; that is, income after taxes. The categories of such expenditures are food, housing, clothing, transportation, savings and financial investments, recreation, personal business, medical care and insurance, education, personal insurance, and miscellaneous. Each of these categories can be subdivided to suit an individual's needs; but a Standard Personal Expenditures Accounting Systems (SPEAS) is essential to the acquisition of useful data.

To illustrate the point compare the expenditure rankings of the two following individuals.

Case I

Account	%
Food	27.8
Housing	27.6
Transportation	15.7
Clothing	11.9
Medical care	6.3
Personal insurance	5.1
Recreation	4.3
Personal business	0.7
Miscellaneous	0.6
Savings & Inv.	0.0
Education	0.0

The Standard Personal Expenditures Accounting Systems (SPEAS)

FOOD
011 Non-taxed food items consumed at home
012 Taxed food items
013 Dining out

HOUSING
021 Rent
022 Home ownership (purchase & financing, maintenance & repairs, commodities, services)
023 Fuel & utilities (fuel oil, coal, gas, electricity, other utilities)
024 Household furnishing (textiles, furniture, floor coverings, appliances, other durable house furnishings)
025 Household operation (housekeeping supplies, housekeeping services)
026 Personal property insurance
027 Hotel & motel bills
028 Telephone

CLOTHING
031 Men's apparel
032 Boy's apparel
033 Women's apparel
034 Girl's apparel
035 Jewelry
036 Cleaning & repair of clothing
037 Personal care (toilet goods, beauty & barber shop services)
038 Luggage

TRANSPORTATION
041 Purchase of automobile
042 Gasoline
043 Auto parts
044 Auto repair
045 Auto maintenance (oil, washing, antifreeze, etc.)
046 Parking
047 Tolls
048 Auto insurance
049 Purchased transportation (buses, taxicabs, trains, airplanes, etc.)

SAVINGS & INVESTMENTS
051 Savings deposits
052 Short-term investments
053 Long-term investments

RECREATION
061 Recreational goods (sports equipment, games, etc.)
062 Recreational services (entrance fees, etc.)
063 Subscriptions for magazines
064 Books
065 Tobacco products
066 Alcoholic beverages
067 Non-medical drugs

PERSONAL BUSINESS
071 Postage
072 Services (printing, etc.)
073 Supplies
074 Equipment
075 Brokerage charges & investment counseling
076 Legal services
077 Bank services
078 Expenses of handling life insurance
079 Funeral & burial expenses

MEDICAL CARE & INSURANCE
081 Professional medical services
082 Prescriptions & drugs
083 Hospital services
084 Health insurance
085 Ophthalmic products
086 Orthopedic appliances

EDUCATION
091 Tuition & fees for formal ed.
092 Musical & other instruction
093 Textbooks & supplies

PERSONAL INSURANCE
101 Life insurance

MISCELLANEOUS
111 Alimony
112 Gambling
113 Support payments
114 Gifts
115 Sales tax
116 Miscellaneous

Case II

Account	%
Savings & Inv.	23.0
Housing	19.1
Food	14.0
Clothing	9.3
Transportation	9.2
Miscellaneous	7.8
Personal insurance	5.3
Medical care	3.9
Recreation	3.5
Personal business	3.1
Education	1.8

Note that Case I uses 89.3% of his funds for basic living needs and none for savings and investments while Case II uses 55.5% for the basic needs and 30.8% for savings, investments, and miscellaneous. The two rankings illustrate two different budget strategies. Case I can appropriately be called a "Basic Needs Strategy" and Case II an "Increase Net Worth Strategy"; these terms are descriptive without knowledge of the actual amount of income spent by the two individuals.

The SPEAS subdivides the eleven major accounts to provide an accommodation for every possible personal expenditure. Food is the first account and all subdivisions of this expenditure category are indicated by the numerals "01" as the first digit of the account number. Thus 011 represents the account for all non-taxed food items consumed at home, 012 all taxed food items, and 013 all eating-away-from-home expenses. If an individual wanted a further breakdown of his expenses he could let 011.1 represent dairy items, 011.2 meat, 011.3 fruits and vegetables, 011.4 cereals, 011.5 processed foods,-..., 012.1 soft drinks, 013.2 candy ,..., 013.2 menu items, 013.2 tips, etc. Such subdivisions would amount to customized systems based on the SPEAS.

The SPEAS presented here has been based on expenditure categories as itemized by the "Groups of Goods and Services Priced for the Consumer Price Index" by the United States Bureau of Labor Statistics, the "Expenditure of Income" tables listed in *Social Indicators 1973* by the Statistical Policy Division of the United States Office of Management and Budget, *Publication 17, Your Federal Income Tax* published by the Internal Revenue Service, and the results of experimental implementation.

Indicators of Performance

This system makes it easy to account for any expenditure. Some account expenditures will occur frequently; that is, during any typical week, and the user will soon become so familiar with them that it will not be necessary to check the SPEAS listing to identify the account number. In fact, the user may soon come to refer to his more common expenditures by their SPEAS number; for example, expenditures for non-taxed food consumed in the home might be referred to as "zero eleven expenditures". Other expenditures occur monthly, such as 021 payments (rent). Some may never be used by any given individual; for example, a homeowner will not use account 021, as his monthly expenditures for housing will be under 022 (home ownership). The pattern of use may even serve as an indication of an individual's socio-economic status (SES) and the investigation of such relationships is interesting from a sociological point of view (socio-economic status is determined by parameters like annual income, occupation, age, sex, and marital status). Within certain SES strata, the variations in expenditure patterns could be interpreted as indications of the personalities of individuals which suggest hypotheses interesting from a psychological point of view; that is, you could analyze someone's personality in terms of the things that he buys.

Such investigations would be searching for key indicators as measures of an individual's performance just as batting averages, home runs and runs batted in have come to be used as key indicators of a baseball player's performance as a hitter. Individuals, in turn, would be interested in assessing their financial behaviors in comparison with well-documented norms. The norms, together with measures of an individual's performance, would indicate what goals are realistic and what goals are beyond the limits of possibility. Anyone knowledgeable about baseball, for

example, would recognize a major leaguer's goals of batting .500, hitting 90 home runs, and batting in 400 runs as impossible to attain. It may be just as impossible for the individual described in Case I to put 20% of his income into savings and financial investments.

It would not be possible to compute a hitter's average in the first place unless there were standard agreements about how to interpret the event of a hit ball. The average is affected by whether the event is judged to be a hit, an out, a safety by error, a fielder's choice, or a sacrifice. These categories indicate the standard accounting system of baseball hitting. Once the definitions are agreed upon, it is possible to compute averages and to identify who is the batting champion by simply seeing who has the highest average. It is also possible to compute the average of all the hitters' averages to establish a performance norm. Such norms are appropriate for hitters in the same league and the norms will differ from little league to amateur adult league, and from minor pro league to major pro league. The different leagues are analogous to different socio-economic strata in the economy.

Given that a person is in a specific SES, say a full-time college student, a blue-collar homeowner with children, a professional married couple who rent an apartment in a large city, etc., the pattern of his expenditures gives an objective analysis of his personality. His expenditures will show more than anything else what he really is interested in and what he wants to do or to have. A question like "How closely do his *ideas* about what he wants match his pattern of spending?" could be answered directly. "How intelligently does he go after what he wants?" would be measured by determining how much money he is wasting, and money wastage could be measured if norms for people in similar circumstances were known.

Many people are interested in monitoring their financial behaviors out of sheer curiosity about themselves and a simple desire to know exactly how they are spending their money.

Rules

In accounting for personal expenditures, income for the sale of assets should be debited to the appropriate account; that is, entered as a negative expense. For example, if one sells some personal furniture, the amount of money received should be subtracted from account 024 (household furnishings). Business expenses need not be recorded to the extent that they are to be reimbursed by a company. Expenditures should be recorded according to the intent of the purchase; for example, automobile antifreeze purchased for one's car but later given to a friend is recorded under 045 (auto maintenance), not under 114 (gifts). Sales-tax charges should be separated from all goods and services taxed and entered in account 115.

The manner in which expenditures are to be recorded depends on the device used. Since a microcomputer is not handy enough to carry around in the pocket, it is sufficient to collect receipts for expenditures and enter the expenditures when convenient. Receipts are available for practically everything except parking meters; checks and credit card receipts facilitate this procedure. If an individual wanted to record every minor expenditure such as parking meter costs he could use some form of petty cash voucher to make the notation. The BASIC program composed by John G. Donohue of the Computer Shop of Syracuse and included in this article is designed to record, update, compute percentage, and rank the expenditures on a microcomputer.

Design for a SPEAS Calculator

Since the rapid and continuing decrease in component prices provides the hardware capability for consumer applications such as financial behaviorism, and the hardware is relatively easy to do, the breakthroughs are in such innovative application designs as this article proposes. Hand-held computers manufactured to perform the SPEAS recordings and strategy computations are quite feasible and the industry has expressed interest in such a development. Such a device would eliminate the necessity to collect receipts, facilitate the immediate recording of all expenditures, and expand the use of financial behavioristic methods to a larger portion of the population. It would feature a memory that retains the data when the device is turned off as the check-balancing calculators presently do, a key to review the day's entries to check for any obvious mistakes (such as an entry of $3,000 for parking), and an accumulator key for summation of the day's accounts into the monthly accumulation of expenditures. At the end of the month the results could then be stored in a

```
10 REM              ************************
20 REM              * FINANCIAL BEHAVIORISM *
30 REM        ******* --UPDATING EXPENSES-- *******
40 REM        *   PROGRAMMED BY JOHN G. DONOHUE   *
50 REM        * FOR THE COMPUTER SHOP OF SYRACUSE *
60 REM        *******   NOVEMBER 20, 1977   *******
70 REM              ************************
80 REM
90 REM
100 REM       **                          **
110 REM     *****     INSTRUCTIONS        *****
120 REM       **                          **
130 INPUT "DO YOU WANT INSTRUCTIONS?  ",R$
140 IF R$(1,1)="Y" THEN GOSUB 1060
150 REM       **                                **
160 REM     *****     VARIABLES INITIALIZED     *****
170 REM       **                                **
180 P=5                       \REM MY DEVICE #5 IS PRINTER
190 N=63                      \REM NUMBER OF ACCOUNTS YOU ARE USING
200 DATA 011,012,013          \REM THE ACCOUNT NUMBERS
210 DATA 021,022,023,024,025,026,027,028
220 DATA 031,032,033,034,035,036,037,038
230 DATA 041,042,043,044,045,046,047,048,049
240 DATA 051,052,053
250 DATA 061,062,063,064,065,066,067
260 DATA 071,072,073,074,075,076,077,078,079
270 DATA 081,082,083,084,085,086
280 DATA 091,092,093
290 DATA 101
300 DATA 111,112,113,114,115,116
310 OPEN #0,"FINBEH"
320 DIM A(N),B(N)
330 FOR I=1TON
340 READ A(I)
350 NEXT I
360 INPUT "IS FILE INITIALIZED? ",R$
370 IF R$(1,1)<>"Y" THEN GOSUB 970
380 REM       **                          **
390 REM     *****     ASK QUESTIONS       *****
400 REM       **                          **
```

microcomputer disk or a hand-written record could be kept.

Needed Software Developments

Needed developments for applications software are as abundant as the private businesses, government agencies, and academic institutions that conduct research into consumer behaviors. Consider for example:

- The type of research conducted by the Nielsen Company and the National Opinion Research Center; the marketing and advertising industries have long been searching for some way to develop accurate correlations between what people *say* they do or do not like and what they actually buy.
- The expenditure of income tables published in *Social Indicators 1973* by the United State Department of Commerce bases its figures for "Consumer unit expenditures, by type of product and service for selected income groups" on data supplied by industry with little input from "consumer units" themselves. (Consumer units include (1) groups of people living together who pooled their incomes and drew from a common fund for their major items of expense and (2) persons living alone or in a household with others but who are financially independent.) This book is the first published statistical analysis that attempts to develop indicators that reveal not only the status of the population in relation to a perceived social objective, but that also furnish some idea of what forces were influencing that status. "At the present time," the introduction states, "not enough is known about the cause and effect of social conditions to develop such ideal indicators." Now that the instruments have been developed to acquire the data descriptive of the financial behaviors of consumer units the feasibility of developing ideal social indicators is less remote.
- The "Review of Applied Urban Research" published by the Center for Applied Urban Research at the University of Nebraska at Omaha is an index of the relative attractiveness of 100 cities. The factors used to calculate a ranking of the cities include things like the "cost of eating out as percent of per capita daily income". The information gathered to compose the index would be enhanced by input from studies of the actual financial behaviors of statistically significant populations of the cities.

Theoretical and Practical Developments

The conceptual structure of financial behaviorism itself is based on the fundamental theorems of information theory. These theorems rely on the mathematical description of entropy to calculate the information content of

```
410 PRINT\PRINT"FINANCIAL BEHAVIORISM"\PRINT"--------- ----------"
420 PRINT "TO CLOSE FILE AND END PROGRAM, TYPE '0,0'."
430 INPUT "ENTER ACCOUNT NUMBER (COMMA) EXPENDITURE: ",X,Y
440 IF X=0 THEN 650
450 FOR I=1TON
460 IF X=A(I) THEN EXIT 510  \REM FIND I OF THIS ACCT.#
470 NEXT I
480 PRINT "ACCOUNT NUMBER IS IN ERROR.  TRY AGAIN --"\PRINT
490 GOTO 430
500 REM       **                          **
510 REM    *****     WRITE REPLIES        *****
520 REM       **                          **
530 READ #0%(I-1)*15, A,B
540 IF A=X THEN 620          \REM DOUBLE CHECK
550 PRINT "MISMATCH OF FILES WITH DATA STATEMENT"
560 CLOSE#0\OPEN#0,"FINBEH"
570 FOR I=1 TO N
580 WRITE #0%(I-1)*15+10,0,NOENDMARK
590 NEXT I
600 CLOSE#0
610 END
620 WRITE #0%(I-1)*15+5,B+Y,NOENDMARK
630 CLOSE#0\OPEN#0,"FINBEH"
640 GOTO 430
650 REM       **                                          **
660 REM    *****     RANK THEM AND PLACE % ON FILE        *****
670 REM       **                                          **
680 CLOSE#0\OPEN#0,"FINBEH"
690 T=0
700 FOR I=1 TO N
710 READ #0,A,B(I)\WRITE#0,0,NOENDMARK
720 T=T+B(I)
730 NEXT I                   \REM GET TOTAL EXPENDITURES T
740 CLOSE#0\OPEN#0,"FINBEH"
750 FOR I=1 TO N
760 WRITE #0%(I-1)*15+10,B(I)/T*100,NOENDMARK
770 NEXT I                   \REM SAVE PERCENTAGES ON FILE
780 FOR J=1 TO 5\PRINT#P,\NEXT J
790 PRINT #P,"FINANCIAL BEHAVIORISM -- EXPENDITURES"
800 PRINT #P,
810 PRINT #P,"   TOTAL: $",%8F2,T
820 PRINT #P,
830 PRINT #P,"ACCT#    EXPENDITURE    % OF TOTAL"
840 C1=-1
850 FOR I=1 TO N             \REM RANK THEM AND PRINT IN ORDER
860 READ #0%(I-1)*15, A,B,C
870 IF C>C1 THEN I1=I
880 IF C>C1 THEN C1=C
890 NEXT I
900 IF C1>-1 THEN 930
910 FOR J=1 TO 5\PRINT#P,\NEXT J
920 GOTO 560
930 PRINT #P,"#",%4I,A(I1),"     $",%8F2,B(I1),"       ",%5F1,C1,"%"
940 WRITE #0%(I1-1)*15+10,-1,NOENDMARK
950 GOTO 840
960 REM-------------------------------------------------------
970 REM       **                          **
980 REM    *****     INITIALIZE THE FILE  *****
990 REM       **                          **
1000 REM
1010 FOR I=1TON
1020 WRITE #0,A(I),0,0
1030 NEXT I
1040 CLOSE#0\OPEN#0,"FINBEH"
1050 RETURN
1060 REM      **                          **
1070 REM   *****     INSTRUCTIONS         *****
1080 REM      **                          **
1090 PRINT "THIS PROGRAM IS WRITTEN IN NORTH-STAR BASIC TO HELP"
1100 PRINT "KEEP TRACK OF SUBTOTALS OF EXPENDITURES IN EACH OF"
1110 PRINT "'N' CATEGORIES.  EACH CATEGORY HAS AN 'ACCOUNT #' A(I),"
1120 PRINT "AND A TYPICAL RECORD ON THE DISK CONTAINS THREE FIELDS:"
1130 PRINT "    THE ACCOUNT NUMBER A(I) OR A"
1140 PRINT "    THE CURRENT SUBTOTAL OF EXPENDITURES IN THAT ACCT."
1150 PRINT "        CALLED B (OR B+Y WHEN UPDATED)."
1160 PRINT "    THE PERCENTAGE THAT SUBTOTAL REPRESENTS OUT OF"
1170 PRINT "        THE TOTAL EXPENDITURES."
1180 PRINT
1190 PRINT "THIS PROGRAM MAY BE USED AT THE END OF EACH DAY TO"
1200 PRINT "UPDATE YOUR RECORD OF EXPENDITURES IN EACH ACCOUNT."
1210 PRINT
1220 PRINT "IT WILL PRINT OUT A LISTING OF EXPENDITURES BY"
1230 PRINT "ACCOUNT, RANKED BY % OF TOTAL."
1240 RETURN
```

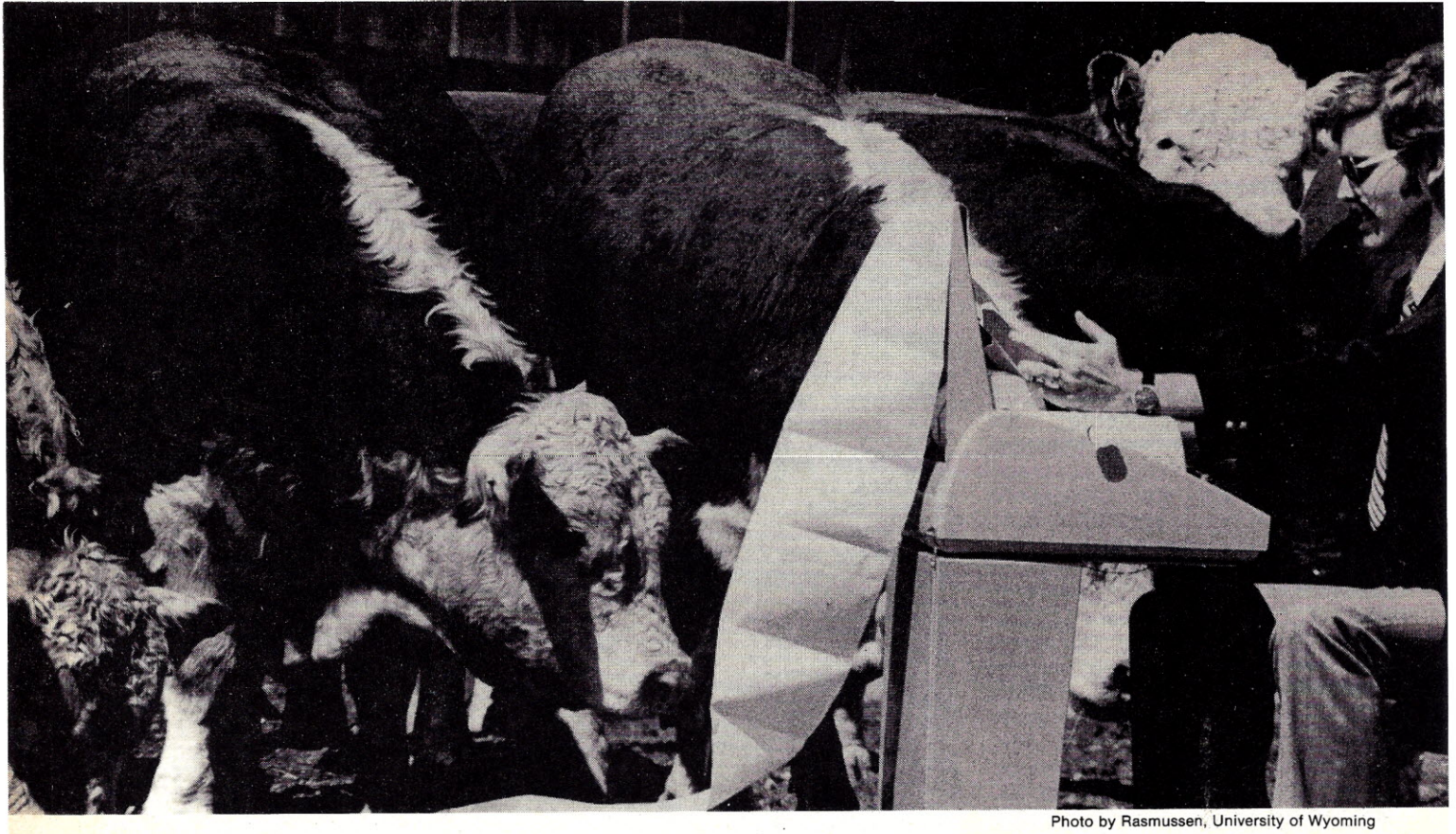


Photo by Rasmussen, University of Wyoming

GET THE MOST OUT OF YOUR COMPUTER WITH CREATIVE COMPUTING MAGAZINE

No computer magazine gives you more applications than we do! Games. Puzzles. Sports simulations. CAI. Computer art. Artificial intelligence. Needlepoint. Music and speech synthesis. Investment analysis. You name it. We've got it. And that's just the beginning!

Whatever your access to computer power—home computer kit, mini, time-sharing terminal—*Creative Computing* is on your wavelength. Whatever your computer application—recreation, education, business, household management, even building control—*Creative Computing* speaks your language.

Read through pages of thoroughly documented programs with complete listings and sample runs. All made easy for you to use. Learn about everything from new software to microprocessors to new uses for home computers. And all in simple, understandable terms. And there's still more. *Creative Computing* discusses creative programming techniques like sort algorithms, shuffling and string manipulation to make your own programming easier and more efficient.

We can even save you time and money. Our extensive resource section is filled with all kinds of facts plus evaluations of hundreds of items. Including microcomputers, terminals, peripherals, software packages, periodicals, booklets and dealers. We also give you no-nonsense equipment profiles to help you decide which computer is best for you—before you spend money on one that isn't.

We've got fiction too. From the best authors in the field, like Asimov, Pohl and Clarke. Plus timely reviews of computer books, vendor manuals and government pamphlets. And so much more!

Isn't it about time you subscribed to *Creative Computing*? It's the smart way to get the most out of your computer.

Complete this coupon and mail it today. Or for fast response, call our toll-free hot line.

800-631-8112

(In New Jersey call (201) 540-0445)

I want to get the most out of my computer.
Please enter my subscription to:

creative computing

Term	USA	Foreign Surface	Foreign Air
☐ 1-year	☐ $ 8	☐ $ 12	☐ $ 20
☐ 2-year	☐ $ 15	☐ $ 23	☐ $ 39
☐ 3-year	☐ $ 21	☐ $ 33	☐ $ 57
☐ Lifetime	☐ $300	☐ $400	☐ $600
☐ Vol. 1 Bound	☐ $ 10	☐ $ 12	☐ $ 15
☐ Vol. 2 Bound	☐ $ 10	☐ $ 12	☐ $ 15

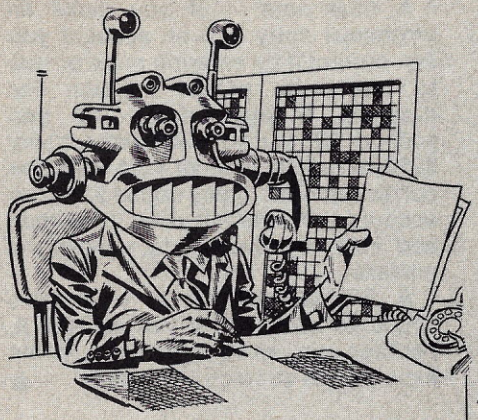
☐ Payment Enclosed
☐ Visa/Bank Americard ☐ Master Charge
Card No. _____
Expiration date _____
☐ Please bill me ($1.00 billing fee will be added; foreign and book orders must be prepaid)

Name _____
Address _____
City _____ State _____ Zip _____

Send to: Creative Computing Attn: Sara
P.O. Box 789-M, Morristown, N.J. 07960

CIRCLE 124 ON READER SERVICE CARD

creative computing brings you its best!



Creative Computing Magazine

The #1 computer applications magazine helps you get the most out of your computer, be it a micro, mini or timesharing system. Plenty of practical applications in education, small business management, household management, recreation and games, and building control. Programming techniques, for those writing their own, include sort techniques, recursion, and file structures. Complete listings and sample runs of games, investment analysis, music synthesis, artificial intelligence programs and much more. Articles, in-depth book and resource reviews, hardware and software evaluations and even some fiction and foolishness. "But the best thing about *Creative Computing* is its feisty, friendly attitude" — *The Village Voice.* Subscriptions, USA 1 year $8 [1A], 3 years $21 [1B]; foreign 1 year $12 [1G], 3 years $33 [1H]. Sample copy $2 [1C]. Volume 1 and 2 issues have been edited into two big 324-page books. *American Vocational Journal* said of Volume 1, "This book is the 'Whole Earth Catalog' of computers." $8.95 [6A]. Volume 2 continues in the same tradition, "Non-technical in approach, its pages are filled with information, articles, games, and activities. Fun layout." — *American Libraries.* $8.95 [6B]. Also available is a special package of all the single issues in Volume 3 for only $8 [1N].

Some Common BASIC Programs

Adam Osborne. An ideal workbook to accompany a BASIC programming course or for one's own use. Contains 76 general purpose, practical BASIC programs with complete listings. 192 pp. $7.50 [7M].

Game Playing With Computers

Donald Spencer. A good mind-exercising resource. Contains over 70 games, puzzles and recreations for the computer. The games, in both BASIC and FORTRAN, have detailed descriptions, flowcharts and output. Also includes a history of game-playing machines and lots of "how-to" write your own games. 320 pp. hardbound $16.95 [8S].

BASIC Programming

Kemeny and *Kurtz.* An excellent gradual introduction to computer programming in BASIC with lots of samples ranging from simple to complex. Rated "The best text on BASIC on almost all counts." 150 pp. hardbound $8.50 [7E].

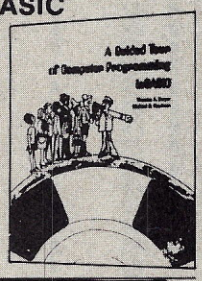
Byte Magazine

If you're considering your own personal computer, then you need *Byte.* With an emphasis on hardware, *Byte* provides a wealth of information on how to get started at an affordable price. Theory, opinion, how-to articles. Subscriptions, USA 1 year $12 [2A], 3 years $32 [2B]; foreign 1 year $25 [2F]. First issues of this now classic publication have been edited into *The Best of Byte, Volume 1.* 386 pp. $11.95 [6F].

Microcomputer Design

Donald P. Martin. This book is well-suited for the engineer who's designing microprocessors into his company's products. Not just block diagrams or vague theory, but dozens of practical circuits with schematics for CPUs based on 8008 chips. Includes interfacing to A/D, D/A, LED digits, UARTs, teletypewriters. Over 400 pp $14.95 [9P].

A Guided Tour of Computer Programming In BASIC



Dyer and *Kaufman.* This book tops all introductory texts on BASIC. Filled with detail and examples, it includes sample programs for many simulations, several games, reservations systems and payroll. Aimed at the novice, but of value to everyone. 156 pp. $4.80 [8L].

Artist And Computer

"Get yourselves a copy of this book if you enjoy feeding your mind a diet of tantalizing high-impact information." — *San Francisco Review of Books.* In this book, 35 artists present a multitude of computer uses and the very latest techniques in computer-generated art. 132 pp. $4.95 softbound, [6D], $10 hardbound [6E].

BASIC Computer Games: Microcomputer Edition

David H. Ahl. An anthology of 101 games and simulations. Everything from Acey-Deucey to Yahtzee, all in BASIC. The only book of its kind with complete listings, sample runs and descriptive write-up of each game. 248 pp. $7.50 [6C].

Creative Computing Catalog

FREE 16-page reference to computer-related products not readily available in commercial stores. Lists books, magazines, T-shirts, game, binary dice and more [5A

The Colossal Computer Cartoon Book

Sit back and relax. Take a break with the best collection of computer cartoons ever. Popular gift item, too. 128 pp. $4.95 [6G].

Problem Solving With The Computer

Ted Sage. Used in conjunction with the traditional high school math curriculum, this book stresses problem analysis in algebra and geometry. This is the most widely adopted text in computer mathematics. 244 pp. $6.95 [8J].

Computer Rage

Fun and educational new board game based on a large-scale multi-processing computer system. The object is to move your three programs from input to output. Moves are determined by the roll of three binary dice representing bits. Hazards include priority interrupts, program bugs, decision symbols, power failures and restricted input and output channels. Notes included for adapting game for school instruction. Great gift item. Ages 8-adult, 2-4 players. $8.95 [6H].

Getting Involved With Your Own Computer

Solomon and *Viet.* One of the first books on microcomputers that requires no previous knowledge of electronics or computer programming. Tells you where to find information, explains basic concepts and summarizes existing systems. Good place for the neophyte to begin. 216 pp. $5.95 [9N].

Fun & Games With The Computer

Ted Sage. "This book is designed as a text for a one-semester course in computer programming using the BASIC language. The programs used as illustrations and exercises are games rather than mathematical algorithms, in order to make the book appealing and accessible to more students. The text is well written, with many excellent sample programs. Highly recommended." — *The Mathematics Teacher.* 351 pp. $6.95 [8B].

An Introduction to Microcomputers

Adam Osborne. Volume 0, "The Beginner's Book" covers microcomputer system components, how they work together; number systems, the basics of programming, and putting it all together in a system of your own. 264 pp. $7.50 [9T]. Volume 1, Basic Concepts, also assumes no prior knowledge of computers. It covers basic principles, binary arithmetic, the microprocessor CPU, I/O logic, memory organization and programming. 264 pp. $7.50 [9K]. Volume 2 (2nd Edition), Some Real Products, covers 20 actual microprocessors in considerable detail including timing diagrams, instruction sets, and interrupts. 760 pp. $15 [9L].

For faster service call toll free

800-631-8112

In NJ call **201-540-0445**

creative computing P.O. Box 789M, Morristown, NJ 07960 USA

Please send me the following:

Quan.	Cat.	Description	Price

Books shipping charge
$1 USA,$2 Foreign _____
NJ residents add 5% Sales tax _____
8 GHJ TOTAL _____

Name _____
Address _____
City _____ State _____ Zip _____
☐ Cash, check, M.O. enclosed
☐ Bank Americard/VISA
☐ Master Charge
Card No. _____
Expiration date _____

events that could possibly be a message. The expenditure transactions of any individual provide the data that describe the messages sent back and forth between the individual and the economy with money functioning as the measure of each one's communication with the other. Financial behavioristic data can be analyzed in terms of the mathematical models used to describe the communication capabilities of electronic devices; such theoretical explorations would seek to develop meaningful interpretations of various patterns of financial behaviors so that more efficient money systems can be designed.

The strategies exemplified as Case I and Case II, the Basic Needs Strategy and the Increase Net Worth Strategy, are representations derived from the income expenditure tables of *Social Indicators 1973.* Case I represents an individual with an after-tax income of $5,000 and who probably supports a family of about five. Case II represents an individual with an after-tax income of $15,000 or more and who probably supports a family of two or three. (1960 dollars: Data presented in the table are taken from the Survey of Consumer Expenditures, 1960-1961, a sample survey of representative consumer units in the United States conducted by the Bureau of Labor Statistics.) The individual described by Case I is not really free to adopt the Increase Net Worth Strategy; daily survival needs structure his socio-economic environment and limit his strategy options. But a ranking of his expenditure percents by the more detailed SPEAS can reveal options that could realistically indicate more efficient uses of his money to accomplish his objectives. The same is true of Case II. Moreover, sufficient data would make it feasible to develop reality-based simulations of the effects on the individual's budget of having another child, relocating, buying a house, etc. Projections describing the financing of such events would be interesting not only to the individual but to loan officers as well.

A financial behaviorism strategy has been defined simply as the ranking of an individual's account expenditures by percent. The number of optional rankings realistically available to any individual is an indicator descriptive of that individual's financial maneuverability. The number of significantly different options available to any individual provides the key to a classification system. To determine what options are significantly different it is necessary to calculate the mean and standard deviation of the individual's account percentages and determine whether any account in the upper range can be switched with an account in the middle or lower range. For example, could Case I switch transportation (15.7%) with personal business (0.7%)? Probably not; his transportation is likely to be a survival need. Could Case II switch savings and financial investments (23%) with recreation (3.5%)? Probably; he could decide to use the money customarily put into savings, stocks and bonds to take a cruise vacation next month.

A handy indicator is the amount of sales tax (account 115) spent by the individual. Sales-tax entries can be accumulated by amount as well as by percent for comparison with the values listed in the "Optional Sales Tax Tables" of the Internal Revenue Service. For example, the deductible amount listed for a family of over five with an income range of six to seven thousand dollars is $109 for the year, or about $9 per month (in New York State; other States are included in the table of *Publication 17, Your Federal Income Tax),* and for a family of three with an income range of fifteen to sixteen thousand dollars it is $178, or about $15 per month. With the tax-table values stored in the microcomputer, an individual can have the ratio of his sales-tax expenditure to the table figure calculated and such ratios can serve as indicators of the individual's participation in the marketplace economy. Meaningful interpretations of such ratios depend on correlations with other factors as developed through methodical investigations of proposed hypotheses.

Microcomputer programs that compute and display SPEAS options representative of valid budget alternatives need to be developed; it is proposed here that individuals interested in practicing financial behaviorism develop such programs based on their own experience. Suitable programs will become more widely useful as data banks are developed and a larger portion of the population wants to know how it can use the microcomputer to figure out budget strategies that are realistic in terms of disposable income and efficient in terms of buying desired products and services.

Conclusion

To begin the analysis of an individual's financial behaviors with a microcomputer, the following program has been composed by John G. Donohue of the Computer Shop of Syracuse, New York. It is designed to accumulate expenditures by account, change them to percents of total expenditure, and rank the percents by account. This process reveals the strategy actually used by the person analyzed and is the first step toward investigating the feasibility of realistic budget alternatives.

A final note — if you make an erroneous entry into an account you can correct it by entering more or less to reach the proper amount. For example, if you entered $25 into account 021 and your rent was actually $250 just add in another $225 to get the correct amount. If you added $25 to an account that was not used at all then add in a negative $25 to remove the mistake.

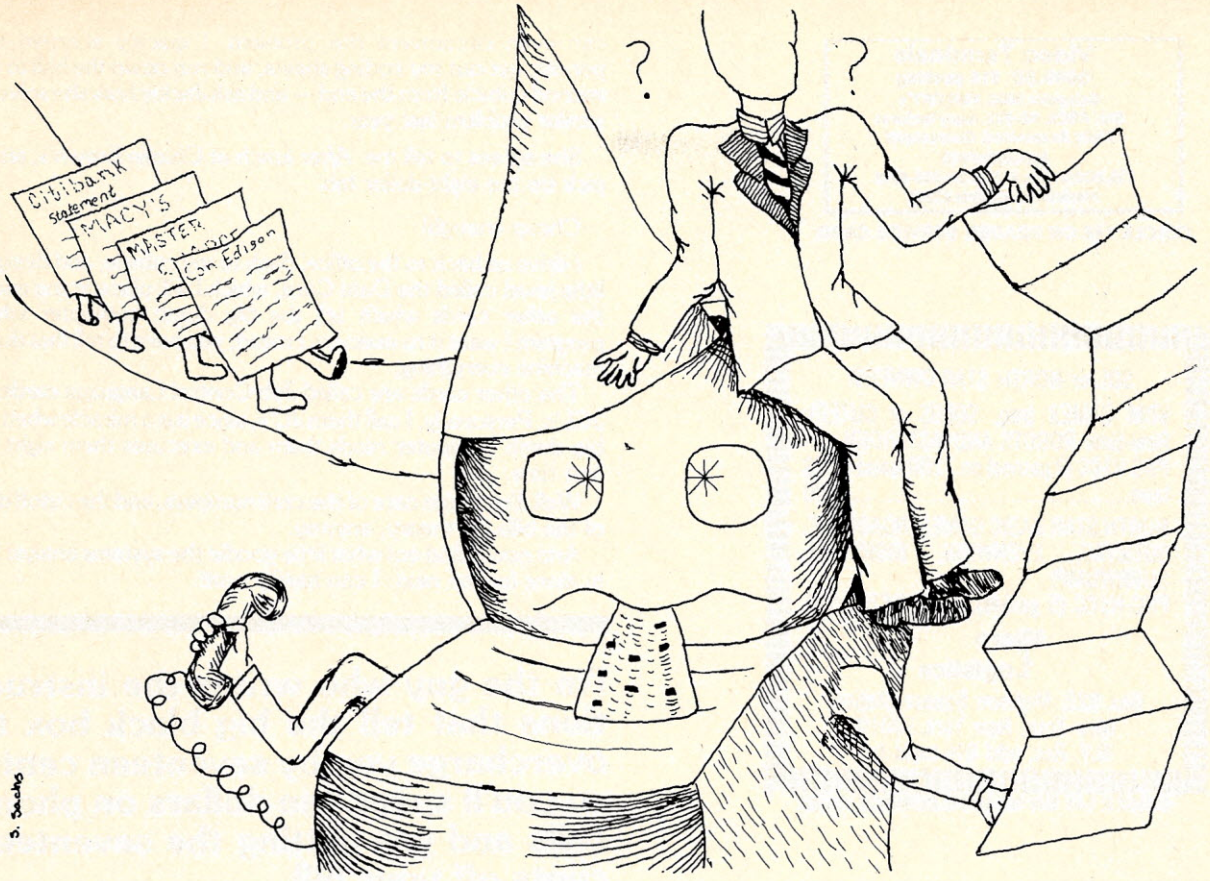
A SPEAS printout of a month's accounts is shown. If you analyze it, perhaps you can think of a name for the expenditure strategy indicated. ■

A SAMPLE PRINTOUT OF A MONTHS EXPENDITURES BY SPEAS ACCOUNTS

FINANCIAL BEHAVIORISM -- EXPENDITURES

TOTAL: $ 827.15

ACCT#	EXPENDITURE	% OF TOTAL
# 21	$ 250.00	30.2%
# 51	$ 115.00	13.9%
# 11	$ 95.00	11.5%
# 101	$ 38.75	4.7%
# 23	$ 32.00	3.9%
# 33	$ 32.00	3.9%
# 81	$ 24.00	2.9%
# 114	$ 24.00	2.9%
# 28	$ 23.00	2.8%
# 42	$ 23.00	2.8%
# 24	$ 17.00	2.1%
# 31	$ 17.00	2.1%
# 115	$ 15.00	1.8%
# 71	$ 13.00	1.6%
# 37	$ 12.00	1.5%
# 13	$ 10.00	1.2%
# 45	$ 10.00	1.2%
# 12	$ 9.00	1.1%
# 72	$ 8.95	1.1%
# 25	$ 8.00	1.0%
# 63	$ 7.80	.9%
# 66	$ 7.25	.9%
# 46	$ 7.00	.8%
# 83	$ 7.00	.8%
# 47	$ 6.00	.7%
# 65	$ 5.40	.7%
# 43	$ 5.00	.6%
# 112	$ 5.00	.6%
# 22	$ .00	.0%
# 26	$ .00	.0%
# 27	$ .00	.0%
# 32	$ .00	.0%
# 34	$ .00	.0%
# 35	$ .00	.0%
# 36	$ .00	.0%
# 38	$ .00	.0%
# 41	$ .00	.0%
# 44	$ .00	.0%
# 48	$ .00	.0%
# 49	$ .00	.0%
# 52	$ .00	.0%
# 53	$ .00	.0%
# 61	$ .00	.0%
# 62	$ .00	.0%
# 64	$ .00	.0%
# 67	$ .00	.0%
# 73	$ .00	.0%
# 74	$ .00	.0%
# 75	$ .00	.0%
# 76	$ .00	.0%
# 77	$ .00	.0%
# 78	$ .00	.0%
# 79	$ .00	.0%
# 82	$ .00	.0%
# 84	$ .00	.0%
# 85	$ .00	.0%
# 86	$ .00	.0%
# 91	$ .00	.0%
# 92	$ .00	.0%
# 93	$ .00	.0%
# 111	$ .00	.0%
# 113	$ .00	.0%
# 116	$ .00	.0%



What's a Programmer?

Wade M. Turner

Do you know what a Computer Programmer is? I do — or rather, I am.

I'm the guy who writes the instructions that tell the big black box to overcharge you by seventeen cents, so you'll spend ten dollars on phone calls and gas getting the seventeen cents off your bill.

Or else I tell the computer not to credit your last payment, and then tack on one-and-a-half percent interest for late charges — and I fix it so the late charges can *never* be adjusted. You will always be indebted to us.

Actually, when you get right down to it, I don't do any of those things. On purpose.

What I do is take some handwritten — or sometimes even *typed,* if he is senior enough — notes from a Systems Analyst, who is next higher in the pecking order around Data Processing shops. Unfortunately, the Systems Analyst comes right out of college into analysis, and knows nothing of the limitations of the instructions a computer is able to interpret and execute. I mean, Computers are *dumb!*

You see all these movies and read books about how smart computers are — forget it. Computers are glorified adding machines and high-speed printers, no more and no less.

You can hook on readers, and boob tubes (Cathode Ray Tubes, or CRTs, as we call them), and voice-response units, male or female — so the customer thinks he's chewing out a nice young lady, when all the time he's talking to a big dumb black box that doesn't understand a word he's saying. But, no matter what you hook on, the computer is still dumb.

Anyhow, I take these notes from the Systems Analyst, telling me to write a program for automatic billing, on ten cycles — every three days — based on the last four digits of the customer's Credit Card Number. Simple so far, right? Wrong!

Our Master Customer File, on thirty reels of tape, which contains all the information about our customers, is arranged alphabetically by name. It has the Credit Card Number in three different places, according to whether it is a straight credit application, a transfer from another state, or a guarantee referral from another customer.

Still with me?

Okay. Even though the numbers are in three different places, and all thirty tapes have to be read every time the program is run, I *can* make the dumb computer look in all three places, or until it finds a number. So that solves that little problem.

Now, I read the note a little further, and I find that the Systems Analyst wants the cutoff date for billing four days previous to the cycle.

Have you ever asked your three-year-old son what time it was? Have you ever told your dog to wash the dishes?

I could probably come up with more apt comparisons, but those should suffice. I mean, the big dumb computer doesn't even know what *year* it is, unless the damn fool operator types the year in on the electric typewriter hooked to the computer and called the Console.

And the less said about operators, the better.

Anyhow. I, with my analytical, spatial-logic-trained mind, and seven years' experience second-guessing Systems Analysts,

A COMPILATION OF FINAGLE'S UNIVERSAL LAWS FOR SYSTEMS ENGINEERS

Axiom #1 In any calculation, any error which can creep in will do so.
Axiom #2 Any error will be in the direction of most harm.
Axiom #3 In any formula, constants (especially from engineering handbooks) are to be treated as variables.
Axiom #4 The best approximation of service conditions in the laboratory will not begin to meet those conditions encountered in actual service.
Axiom #5 The most vital dimension on any plan or drawing stands the greatest chance of being omitted.
Axiom #6 If only one bid can be secured on any project, the price will be unreasonable.
Axiom #7 If a test installation functions perfectly, all susequent production units will malfunction.
Axiom #8 All delivery promises must be multiplied by a factor of 5.0.
Axiom #9 Major changes in construction will always be requested after fabrication is nearly completed.
Axiom #10 Parts that positively cannot be assembled in improper order will be.
Axiom #11 Interchangeable parts won't.
Axiom #12 Manufacturer's specifications of performance should be multiplied by a factor of 0.5.
Axiom #13 Salesmen's claims for performance should be multiplied by a factor of 0.25.
Axiom #14 Installation and Operating Instructions shipped with any device will be promptly discarded by the Receiving Department.
Axiom #15 Any device requiring service or adjustment will be least accessible.
Axiom #16 Service conditions as given on specifications will be exceeded.
Axiom #17 If more than one person is responsible for a miscalculation, no one will be at fault.
Axiom #18 Identical units which test in an identical fashion will not behave in an identical fashion in the field.
Axiom #19 If, in engineering practice, a safety factor is set through service experience at an ultimate value, an ingenious idiot will promptly calculate a method to exceed said safety factor.
Axiom #20 Warranty and guarantee clauses are voided by payment of the invoice.

can even circumvent *this* problem. I quickly sharpen two pencils, get out my coding sheets, and run down the hall to the second cubicle from the end — and ask Jackie how she solved a similar problem last year.

She agrees to tell me. *After* lunch at Charlie Brown's, after I pick up the eight-dollar tab.

Cheap enough!

I drive us back to the office, dash to my cubicle, and design a little jewel called the Date Card, which I will place at the end of the *other* cards which tell the big dumb computer which program I want it to execute. I mean, you have to tell that dumb machine *everything*.

The other cards are called Job Control Language cards, or JCL's. Personally, I call them JC's, since it is a miracle when the big dumb computer reads them and executes them right the first time.

Well. I've taken care of the card numbers, and the cutoff date of the bills. In theory, anyway.

And now, I find out what little goodie the systems analyst has in store for me next. I can hardly wait!

I'm the guy who writes the instructions that tell the big black box to overcharge you by seventeen cents, so you'll spend ten dollars on phone calls and gas getting the seventeen cents off your bill.

I discover that he wants me to list each charge for the month separately, along with the previous payment, the old balance, and the new balance. Nothing to it . . . sixth-grade kids all over the world are doing it — at least they *were*, until some genius came up with New Math. It should be a snap for an intricate, sophisticated conglomerate of integrated microelectronic circuitry that *leases* for sixty thousand bucks a month, and costs in the millions.

And, given the proper instructions, it *is* a snap.

It is a snap, *in theory*. But in practice, the damn fool operator runs my billing program *before* he runs the program that updates the Master Customer File. Therefore, even though Mrs. Solomon from Rolling Hills made a special trip to personally deliver her check to our South Bay store so it could be credited for last month's payment before the deadline, the big dumb computer adds on a one-and-a-half percent late charge for no previous month's payment.

Sound familiar?

When Mrs. Solomon goes to the Credit Department, and threatens to cut up her card and throw it in the Credit Manager's face, all he can do is wear a glassy smile and say, "Sorry, computer error . . ." — and wonder what happened to the good old days, when he could vent his frustration by chewing out the dozen or so girls he kept busy eight hours a day recording charges, licking envelopes, and mailing bills.

I mean, how can you chew out a big dumb black box that won't even cringe? You can't refuse it a raise, or threaten to fire it, or tell it that it can be replaced — you have no ammunition at *all* for a healthy tirade once in a while to keep the old juices flowing.

Sorry. I digressed. Back to the Billing Program.

I have the number, the cutoff date, the previous payment, old and new balance, including itemized charges. You would think that would be more than enough to satisfy any sane, rational human being. And it probably is.

But, it is not enough for my Systems Analyst.

You see, his wife once bought a dress on sale. The ticket his wife saw on the right sleeve of the dress read $14.95, but the ticket the sales clerk read — and wrote the sales slip from — read $16.50. It was on the left sleeve.

The analyst's wife didn't notice the discrepancy until she arrived home and happened to glance at the sales slip — which is *way* too late, as any credit shopper with more than six months' experience will tell you. In fact, in the newer, insta-inventory stores, the minute the sales girl rings the register, it's too late!

So, because of his wife's little problem months and months before, my all-wise Systems Analyst adds a little P.S. on the note, saying allow for credit adjustments up to the cutoff date.

That's easy for *him* to say!

What it means to me, the poor programmer, is one of two things. Either have the letters "CR" added to the money figures on all thirty tapes of the master file — which will automatically increase the master file to thirty-*two tapes* (maybe thirty-*four* tapes around Christmas and Father's Day), or else set up a series of one-letter codes for the keypunch girls to enter wrong — so the whole record gets thrown out by the big dumb computer.

Either way, Mrs. Solomon of Rolling Hills never gets her seventeen cents credit on the Sales Tax the girl overcharged her because the Dorothy Gray Cosmetic, on sale twice a year, had the Excise *and* Sales Tax already included on the paste-on sticker — so the sales girl wouldn't have to keep running her finger down the sales tax indicator each time she sold a Dorothy Gray cosmetic.

And, when Mrs. Solomon confronts the same poor, harried Credit Manager — who still has no one to pick on — all he can do is say, "Sorry, computer error...."

I say, Hogwash!

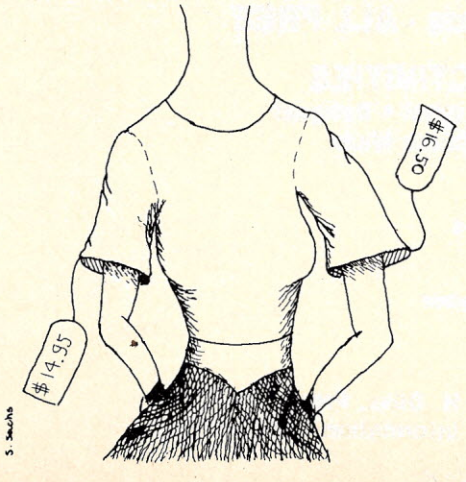
I also say other things, especially when the phone rings at two-thirteen AM — which is the time they always seem to run my programs. At least, the programs that halt with a message to call the programmer because the big, dumb computer didn't understand some perfectly logical instruction, and erased half the Master Customer File, or some silly thing like that.

You see, having a logical mind, and understanding the eccentricities of the big black box, I always have the computer print out what is wrong on the electric typewriter connected to the computer, called the console.

I have it print a three-number code, and then a message. For instance, if the damn fool operator forgets to put the Date Card at the end of the JCL (you remember all that, don't you?), the Console will magically type, "HALT, 101**DATE CARD INVALID OR MISSING***".

Self-explanatory, right? Even a damn fool operator would understand *that*, you would think!

Not the case at all.



The operator automatically reaches for the telephone right beside the Console with his left hand, while he thumbs through the home-phone numbers of programmers assigned to production programs with his right, until he finds my number — which I pay the phone company an extra forty-seven cents a month to keep unlisted, so drunks and fools don't call me at two-thirteen AM.

"Hello ... is this Turner?"

"Mummbbllee"

"This is Dave."

"Dave who?"

"Dave Stern."

I wait ten seconds for him to continue, since I'm almost positive that *he* called *me*. But when he says nothing, I take the bull by the horns. "Well, Dave?"

"Well what?"

"Well, why in the blankety-blank did you wake me up at two o'clock in the blankety-blank morning?"

"Oh, that ... well, it's this program, see ... CB40404."

Again I wait. But he waits longer.

"Well, what about CB40?"

"Oh ... well, it quit running ... but it didn't blow up ... I got a Halt 101 on the console..."

Being the alert, experienced programmer that I am, I knew he didn't mean literally *blow up* — he meant the program came to an abnormal termination and ruined everything the big dumb computer had done to that point.

More importantly, being now wide awake and halfway through my second cigarette, I also sensed a familiar ring to Halt 101.

"Say Dave?"

"Yeah..."

"Was there a message with the Halt 101?"

"Message? Oh...yeah."

"What did the message say, Dave?"

"Say? Oh ... Date ... Card ... In ... Val ... Id ... or missing."

"And what does the Program Run Book say about Halt 101, Dave?"

Since the mid-Sixties, and the inception of the Full Operating System (the big dumb computer that messes up several programs at the same time, instead of one at a time — like the old days), programmers have provided damn fool operators with Program Run Books so the damn fool operators wouldn't wake the poor programmers up at two thirteen AM.

"Program Book? Oh ... I didn't look."

"Well, Dave. I'll save you the trouble ... this time. Halt 101 says the operator didn't insert the Date Card at the end of the JCL ... or else he punched the card wrong ... Are you with me Dave ... Dave?"

"Oh ... yeah. Date Card, huh ... okay ... thanks."

I hang up the phone and run to put on the water for the Instant Yuban. But Dave is too quick for me.

I answer in the kitchen. "Dave ... it's me. Look in the Program Run Book, on the last page, and punch a card exactly like the one there for cycle three..... three, Dave..... one, two, *three!*..... Got it Dave? Dave?"

"Oh ... yeah. Three."

Now, don't get me wrong. Not *all* damn fool operators are like Dave. Nor do *all* Systems Analysts have wives with problems.

However, the great *majority* ... but that's another story.

Besides, I gotta run just now.

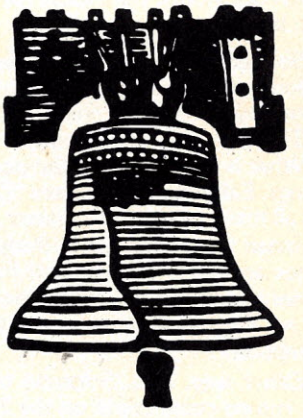
You see, the Systems Analyst gave me a new program this morning, and there's this tricky little formula in it. Nancy had one almost like it about six months ago, so she and I are going to The Second Storey for lunch. I mean, ten dollars is cheap enough, right?

After all, they pay me eighteen thou a year for my knowledge, experience, and ability. I should be entitled to take a nice young lady to lunch once in a while, I would think. ■

Good-bye, Atlantic City

Hello, PHILADELPHIA

PERSONAL COMPUTING '78™



THIRD YEAR — LARGEST YET

PHILADELPHIA CIVIC CENTER
AUGUST 24, 25, 26, 27

FOUR FULL DAYS

August 24

Industry Trade Show

for Dealers, the Industry and Exhibitors Guests **AND** A complete lineup of seminars and meetings.

August 25, 26, 27
PERSONAL COMPUTER SHOW
New Products • Seminars • Forums • Technical Talks
The largest, with 300 booths. The longest, at four days, of any computing show.

Don't Miss

PERSONAL COMPUTING COLLEGE™ • over 80 hours of in-depth seminars - ALL FREE

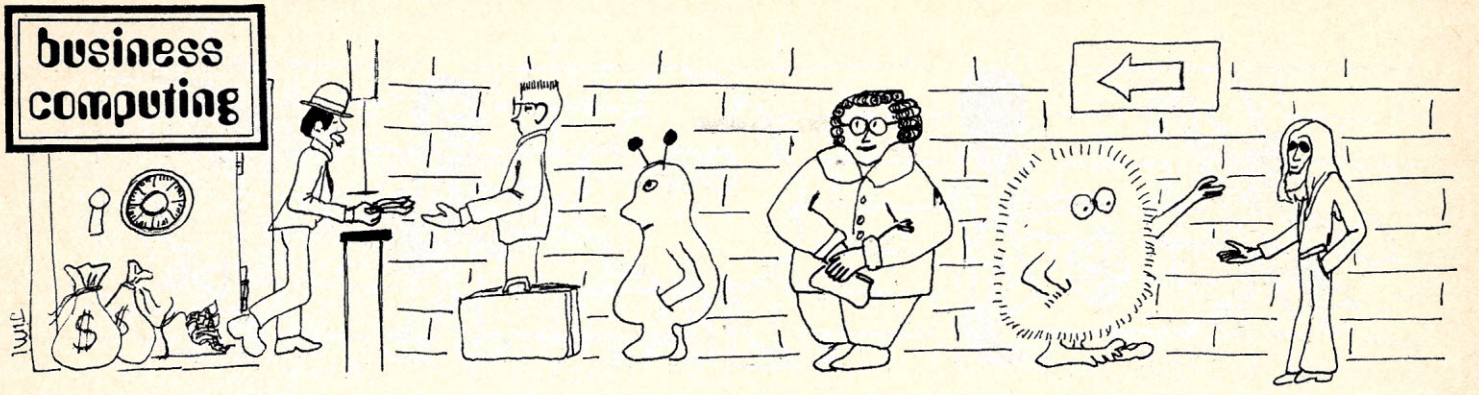
PLUS FUN ACTIVITIES
Art Show • Music Festival • Banquet
Computerized Mouse Maze

Save Money!
Make reservations now!

Exhibitors reserve your space now!

PRODUCED BY
PERSONAL COMPUTING INC. • John H. Dilks, President
Rt. 1 • Box 242 • Mays Landing, NJ 08330 • Information 609-653-1188

CIRCLE 152 ON READER SERVICE CARD



Payroll Overview:
Micros "Meet the Payroll"

Gene Murrow

The payroll... one of the traditional tests of a true businessman's mettle. Pity the inexperienced or untested aspirant to a position of status, whose ideas and worth are discounted with the simple remark, "...yes, but he's never had to meet a payroll."

The most recent claimant to a place in the business world is the microcomputer, and now it finds itself having to perform on payday. Several manufacturers, software designers, and computer outlets have begun advertising "payroll" for the new generation of small computers. Articles in this issue of *Creative Computing* describe some of these systems. But how useful is an automated payroll system? What are the alternatives? What should a *good* payroll system include? How should the astute businessman evaluate the various systems? In this introductory article, we shall provide some answers to these fundamental questions.

Large companies have been using computerized payroll systems for years, and the cost-effectiveness of such systems is taken for granted. The small business of 5 to 75 employees, however, typically employs a manual or semi-automated system, or uses a payroll service provided by a commercial bank. The availability of microcomputer systems challenges the traditional reliance on manual systems or outside payroll services. The price is low, and the power is generally high.

Gene Murrow, Computer Power & Light, Inc., 12321 Ventura Blvd., Studio City, CA 91604.

Compared to automated systems, manual payroll systems have several disadvantages. They require considerable time by a trained bookkeeper or clerk, especially at end-of-quarter and end-of-year reporting times, and therefore are costly. Like any complex procedure which is often carried out by humans under pressure, they are susceptible to error. In addition, they do not easily yield summary data and reports, which might be very useful to management. Most businesses are fortunate if just the basic journals and deduction registers are maintained accurately and kept up to date without the help of an accountant.

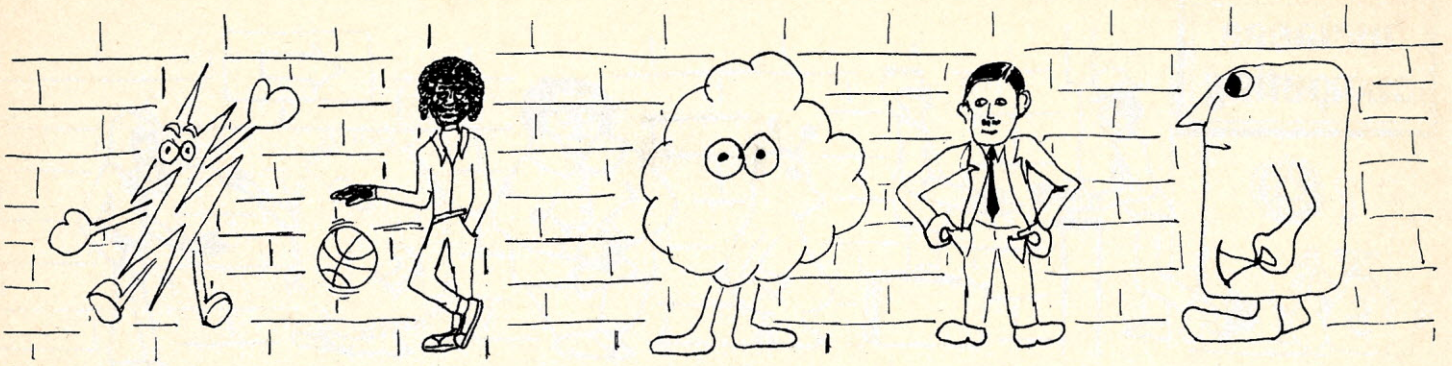
Despite the problems with manual or outside payroll systems, they may well be better than an ill-conceived or unreliable in-house microcomputer system.

The payroll services provided by banks and other service bureaus solve some of these problems, but create new ones of their own. Such systems are generally inflexible, requiring a business to adapt to the system, rather than the reverse. They may not accommodate such desirable features as summary reports by departments within a business, or pay schemes such as draws against commissions. Bookkeepers often spend as much time preparing the data for the outside service bureaus and banks as they would have spent simply filling out the checks and journals themselves. There is often a lag time, or "turnaround time" of several days between the time a change is desired (in pay level, employee status, etc.) and the time such a change is incorporated into the system. Perhaps the biggest disadvantage is the requirement of most banks that enough money to cover a total payroll be kept on deposit at all times. Few small businesses want to keep that amount of cash out of circulation for two weeks or a month.

Despite the problems with manual or outside payroll systems, they may well be better than an ill-conceived or unreliable in-house microcomputer system. Here are some guidelines by which to evaluate a system, based on COMPAL's one and one-half years of experience installing business microcomputers which, among other things, automate the payroll.

Hardware

This should include a printer and a disk drive. The printer produces the reports and registers and prints the paychecks. Without it, your bookkeeper will be doing a lot of copying by hand. The random-access capability of the disk drive allows for easy and flexible maintenance of the data base.



Systems that use a tape cassette instead of a disk will be limited in their ability to make quick updates such as adding or deleting an employe, changing an employe's status, etc. The COMPAL system we recommend to our customers for all accounting functions including payroll includes a 120-line-per-minute printer, and a dual floppy-disk drive storing 315K characters per disk. It leases for $185.38 a month.

Software

A payroll program must actually perform several distinct tasks. These are described below:

- Enter/delete/update employe records: name, address, Soc. Security number; no. of standard deductions; medical and retirement plan "reductions"; pay type (salary amount, hourly wage, commission percentages, or guarantee thresholds).
- Automatically maintain payroll registers: year-to-date and quarter-to-date employe contributions to state and federal taxes; same for employer taxes payable.
- Compute paychecks: efficiently gather hours worked for each employe, commissions earned, vacation and sick days taken, gas allowances, salary advances, other taxable income; then compute gross pay, withholding and deductions, and net pay.
- Print paychecks: allow for form alignment, proper numbering and dating of checks.
- Produce reports: employe and employer contribution registers for each pay period; year-to-date and quarter-to-date summaries; end-of-year W-2's.
- Enable system maintenance: allow modification of salary levels, etc., by privileged personnel only; allow "back-dating" of cumulative records if error detected in current payroll data; allow easy updating of tax computation algorithms when tax laws change; allow (indeed, force) operator to make back-up records easily; allow specification of various management reports (for example, totals by department).

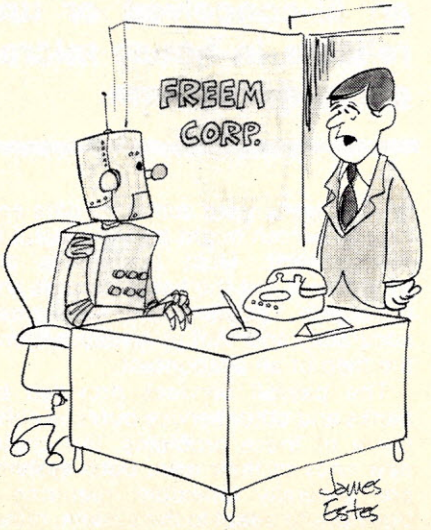
The COMPAL payroll system, offered with the hardware described above, performs all of these jobs and more. A complete payroll involving 25 employes can be run in well under one hour.

Other Considerations

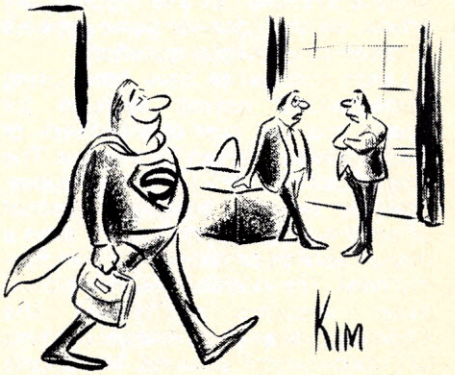
- Maintenance: Who will fix the computer if it breaks on Friday afternoon at 1:30 P.M.? Who will update the program when the tax laws change?
- Training: who will teach your bookkeeper how to run the system? What happens if he leaves in six months?
- Reliability: of the hardware, of the software, and *of the vendor!*

At COMPAL, these last items demand as much of our attention as the others. We must provide instant "loaners" and service to our customers using the payroll system. We train the bookkeeper, the clerk, and anyone else who will be using the system in our own classroom facilities. We see to it that the customer's software is up-to-date and in compliance with the law.

In summary, we advise the businessman not to abandon his common business sense in evaluating a microcomputer-based payroll system. Demand to see the system perform, get references, do the same things you would do in evaluating any addition to your operation. Don't be snowed by the "gee-whiz" aspects of this exciting new industry. Expect to pay for value; "you get what you pay for" does apply to the microcomputer business, as crazy as it sometimes seems. Read the articles in this magazine, ask around, be shrewd, and you will find an electronic associate who can "meet the payroll" as well as you can, and save you a lot of money in the process. ■



"I'd like to apply for a . . . oh, never mind."



"He's been wearing it ever since he single-handedly won that programming contest last week."

business computing

Payroll: Osborne & Associates

The published payroll has been made as general and flexible as possible, without making it unwieldy. A chapter of the book, and various comments throughout, suggest ways to customise the published system to make it best suit your needs.

Mary Borchers
Lon Poole

Osborne & Associates is publishing a series of books providing BASIC source listings and documentation for business data processing. The first book, *Payroll With Cost Accounting,* by Lon Poole and Mary Borchers, is available now at $12.50, and is probably the most complete payroll package published to date.

Payroll will be followed by two more books: *Accounts Payable and Accounts Receivable,* and *General Ledger.* Both of these books are expected to be completed later this year.

These published accounting programs are a direct result of Osborne & Associates' five years of experience serving as software consultants to small and medium-sized businesses. The programs have been tested and updated over the years so that they are now, basically, error-free.

As software consultants, programmers at Osborne & Associates realized that a great deal of diversity exists in accounting methods. Their published payroll has been made as general and flexible as possible, without making it unwieldy. A chapter of the book, and various comments throughout, suggest ways to customize the published system to make it best suit your needs.

Contents

Approximately one-third of the 380-page *Payroll* book is taken by source listings of 35 programs which make up the payroll system. Remarks for each program are included in the margin next to the listing. Line number, variable and special function cross reference tables are also provided for each program.

The remainder of the book is devoted to extensive system and program documentation. A programmer installing the system should read the entire book, but parts of the book intended only for non-programmers are kept separate.

For the programmer there are discussions of program implementation, disk accessing methods, system limitations, printer usage, special printed forms and file layouts. Program-by-program information includes the function of each program, how it works, its limitations, variable usage, CRT display requirements and sample printouts. A program to set up CRT masks is also included.

For the non-programmer, a general Management Guide provides an overview of how the system works. It gives an idea of when to enter data, which program is to follow which, what reports are included and when they should be printed. After reading the Management Guide you are ready to go on to the User's Manual.

The User's Manual gives step-by-step instructions for each of the 35 programs a payroll clerk will use. It includes general data-entry instructions, when and how to use each program, field definitions and their limitations, sample CRT displays with prompt messages, user flowcharts, instructions on error recovery, and what to look for and what to watch out for.

Now you may wonder why there are 35 separate programs.

There are two basic reasons. First, a crude calculation will show that you need approximately 1/35 memory to use the system with 35 programs as compared to the same system written as one huge program.

Second, it allows much more flexibility within the system. Each program performs a specific task. Any program can be run as many times as necessary and in any order, as long as it doesn't affect calculations. For example, nine of the 35 programs are termed "file maintenance" — all they do is allow you to see and change data stored in a file. This is an important feature of this system, because it allows the operator to check and correct data at any time, as many times as necessary, without having to first complete processing (as you might with one huge program). Thus you are able to avoid propagating one error throughout the entire system.

Report programs may also be run as many times as necessary. This makes it easy to get more than one copy of a report, and overcome those uncontrollable bad printouts that often result from the printer itself — the paper does not feed properly, the ribbon gets out of whack, etc.

And even though there are 35 programs, it needn't appear that way. To avoid the tedium of having to load, execute and save each program, there is one program, Menu, whose only function is to load and execute another program which has been selected by the operator. When processing of any other program is complete, it in turn loads and executes the Menu. Thus, to the operator, the system does not appear to the 35 separate programs at all, but one large, multi-faceted program.

Mary Borchers and Lon Poole, Osborne & Associates, P.O. Box 2036, Berkeley, CA 94702.

System Capabilities

The payroll published is based on a biweekly pay period. Up to ten separate companies may be entered, and each company may have up to 9999 employees.

Employees may be classified as one of three types: salary, salary with overtime, or hourly. Earnings are calculated by accumulating any combination of salary, regular hours, overtime hours, vacation pay, holiday pay or piecework pay (except a salary employee may not receive regular hourly pay, and an hourly employee may not be paid a salary).

Taxable or non-taxable lump sums may be added to an employee's total earnings to be included in his paycheck. This is used for paying bonuses, travel reimbursements, etc.

At the end of the pay period, taxes are computed on the total taxable pay, and then deducted from it. U.S. income tax, social security, California state income tax and California disability insurance are automatically withheld according to each employee's deduction claims. Additional federal, state or miscellaneous deductions may also be withheld.

When paychecks are printed, all additional pay and deductions that were included in that check's calculations are detailed on the check stub.

The results of the check calculations are accumulated to monthly, quarterly and yearly totals. Selected information from each paycheck is saved to provide a history of every check issued an employee. These cumulative totals and historical records are the basis for printing the payroll journal, government tax forms (941 and W-2), and other reports. A sample of the payroll journal is shown.

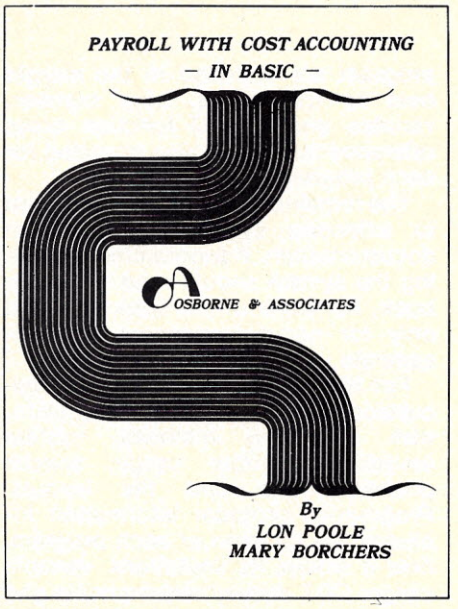
Pay data may be entered at any interval you select — from once a day to once a pay period. As pay data is being entered, it may be associated with a job number. 99999 jobs are allowed for the entire system, with each job subdivided into ten tasks. Subtotals for each task are printed along with job totals on the costing report. Hours charged to each job are also reported by the employee.

Program Development and Implementation

All programs have been written, tested and run on a Wang 2200 series computer using its special extended BASIC and a 5-megabyte disk for data and program storage. It uses 16K bytes of program memory, excluding the BASIC interpreter.

To effectively use this payroll system you will need a CRT display, printer and disk-storage device. Specifically, the CRT should display a minimum of 16 lines by 64-character width, and should be formattable (you can specify cursor position). We suggest a minimum or 16K program memory. Programs are written for a 132-character line width printer; if yours is less, many report programs will need adjustments. A random-access storage device, such as hard or floppy disk, is necessary; sequential-access data storage will simply be too time-consuming. The amount of storage area needed is determined primarily by the number of employees you have.

A programmer is required to set up the system. He will have to set up data files and insure the programs are running properly. If your computer is not compatible with Wang BASIC, the programs will have to be modified to conform to the new BASIC syntax (there is a chapter in the book that describes special features of the Wang Laboratories computer to help you do this).



While you are changing programs to be compatible with your computer, you will probably want to customize them so they match your payroll procedures more exactly. Even though this payroll is general-purpose, some arbitrary decisions had to be made which the authors realize may not be compatible with your payroll methods. To help the programmer do this, a chapter is included on different ways to customize the programs.

Program Conversions

Realizing that publishing programs in Wang BASIC limits the number of businessmen that could use this system, Osborne & Associates is encouraging consultants and individuals to "convert" these programs to other systems and market them. Osborne & Associates charge no license fee for these conversions nor do they demand any royalty on resales. A statement of Osborne & Associates policy on program conversions is reproduced below.

To date, there are three companies in the process of converting this payroll system to other computer systems. Anyone interested in purchasing this payroll for other systems should contact these persons directly:

Alpha-Micro system:
Scott Brim, president
Computer Systems
7952 Secretariat
Las Vegas, NV 89119

Digital Group Z80 and IMSAI (with CP/M disk) systems:
Peter M. Burke
The Basic Business Software Company
P.O. Box 2032
Salt Lake City, UT 84110

Digital Group system:
John Musgrove
Musgrove Engineering
9547 Kindletree Dr.
Houston, TX 77040

In addition, you may purchase machine-readable listings of Wang programs as published in the books by contacting the following:

Wang BASIC on floppy disk:
Richard M. Armour
Atlantic Computing and Consulting, Inc.
1104 Sparrow Road
Chesapeake, VA 23325

Wang BASIC on cassette or hard disk:
Mary Borchers
Osborne & Associates, Inc.
P.O. Box 2036
Berkeley, CA 94702

Any other persons interested in converting the Payroll programs for resale on other computer systems should write to Osborne & Associates and ask for a Statement of Policy and to be included on their referral list.

A Statement of Policy

Osborne & Associates is publishing a series of books providing BASIC source listings and documentation for business data processing programs.

All of the BASIC program books that we have available or currently scheduled copyright the printed word only; they specifically exclude protection of the magnetic surface. This means that we are, in effect, placing the machine-readable form of the software in the public domain while retaining all rights to the human-readable form of the programs. You are free to take any programs out of our books and use, modify or resell them without authorization, royalty or license, but

you cannot give away or sell any portion of the programs in human-readable form. The printed source listings are protected to the last line of readable code.

Does this mean that you must sell a copy of our book with your software? There is no law that we could invoke to force this upon you even if we wished to; however, economics favor that you do so. When you see our books, you will find that the documentation accompanying the programs could not be reproduced by you or anyone else without spending a very large sum of money and wasting a great deal of time. You are thus faced with the option of buying our books wholesale, reselling them retail and making a small profit; or attempting to redo the documentation yourself and taking a substantial loss. Moreover, if you produce your own documentation, it can only include source listings for any new code you add. You cannot reproduce our source listings.

Osborne & Associates is putting itself in the position of supplier to consultants rather than competitor with consultants. Osborne & Associates will not modify programs for any customer, nor do any type of custom programming work. We will, instead, refer all inquiries to consultants. This being the case, we encourage you to tell us what you have done with our programs and what kind of referrals regarding our programs you would like to receive. ■

IMPRESSIVE PRODUCTS — DATE 02/22/78
PAYROLL JOURNAL 02/14/78 - 01/28/78 — PAGE 1

EMP NO. 10095 NAME: SLATS GRUBNICK — SS # 951-00-0000 — STATUS S — DATE/EMP 05/28/74 — PAY RATE 5.0000
ADDR: 10095 MAIN STREET — EMPLOYEE TYPE 1 — FED-EX 1 — CHECK NO. 931 — H&W RATE 0.0000
CHICAGO, CALIFORIA — EMP. CLASS 0 — ST-EX 1 — CHECK DATE 02/12/77 — MON HOURS 85.5000
90095 — INS. CLASS 2 — ADD-EX 0 — VAC HOURS 0.00 — MON PAY 441.2500

	REG HRS	REG PAY	O.T. HRS	O.T. PAY	P.W. HRS	P.W. PAY	H&W PAY	VAC HRS	VAC PAY	OTHER PAY	NON-TAX
CURRENT	80.00	400.00	5.50	41.25	0.00	0.00	0.00	0.00	0.00	0.00	0.00
QTD	240.00	1200.00	10.50	78.75	0.00	0.00	0.00	0.00	0.00	25.00	0.00
YTD	240.00	1200.00	10.50	78.75	0.00	0.00	0.00	0.00	0.00	25.00	0.00

	FED W/H	ST. W/H	F.I.C.A.	S.D.I.	OTHER DED	TOTAL DED	NET PAY	TOTAL PAY
CURRENT	66.75	13.96	25.81	4.41	100.00	210.93	230.32	441.25
QTD	183.87	31.82	70.45	12.03	100.00	398.17	905.58	1303.75
YTD	183.87	31.82	70.45	12.03	100.00	398.17	905.58	1303.75

EMP NO. 10500 NAME: LEOPOLD BLOOM — SS # 101-01-1001 — STATUS H — DATE/EMP 01/24/68 — PAY RATE 1750.0000
ADDR: COMPANY PRESIDENT — EMPLOYEE TYPE 0 — FED-EX 3 — CHECK NO. 932 — H&W RATE 0.0000
10500 ULYSSSES — EMP. CLASS 1 — ST-EX 3 — CHECK DATE 02/12/77 — MON HOURS 82.0000
DUBLIN, CALIFORNIA — INS. CLASS 1 — ADD-EX 1 — VAC HOURS 80.00 — MON PAY 2250.0000

	REG HRS	REG PAY	O.T. HRS	O.T. PAY	P.W. HRS	P.W. PAY	H&W PAY	VAC HRS	VAC PAY	OTHER PAY	NON-TAX
CURRENT	82.00	1750.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	500.00	0.00
QTD	82.00	5250.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	500.00	0.00
YTD	82.00	5250.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	500.00	0.00

	FED W/H	ST. W/H	F.I.C.A.	S.D.I.	OTHER DED	TOTAL DED	NET PAY	TOTAL PAY
CURRENT	695.28	190.56	131.63	22.50	0.00	1039.97	1210.03	2250.00
QTD	1564.04	406.56	336.39	57.50	0.00	2364.49	3385.51	5750.00
YTD	1564.04	406.56	336.39	57.50	0.00	2364.49	3385.51	5750.00

EMP NO. 11229 NAME: TOM JOAD — SS # 001-11-1001 — STATUS S — DATE/EMP 11/12/69 — PAY RATE 512.5000
ADDR: 11229 CANNERY ROW — EMPLOYEE TYPE 0 — FED-EX 0 — CHECK NO. 933 — H&W RATE 0.0000
SALINAS, CALIFORNIA — EMP. CLASS 0 — ST-EX 0 — CHECK DATE 02/12/77 — MON HOURS 88.2500
90345 — INS. CLASS 2 — ADD-EX 0 — VAC HOURS 0.00 — MON PAY 803.3400

	REG HRS	REG PAY	O.T. HRS	O.T. PAY	P.W. HRS	P.W. PAY	H&W PAY	VAC HRS	VAC PAY	OTHER PAY	NON-TAX
CURRENT	40.00	512.50	4.25	40.84	4.00	150.00	0.00	40.00	0.00	100.00	0.00
QTD	40.00	1537.50	4.25	40.84	4.00	150.00	0.00	40.00	0.00	100.00	0.00
YTD	40.00	1537.50	4.25	40.84	4.00	150.00	0.00	40.00	0.00	100.00	0.00

	FED W/H	ST. W/H	F.I.C.A.	S.D.I.	OTHER DED	TOTAL DED	NET PAY	TOTAL PAY
CURRENT	187.20	50.48	47.00	8.03	0.00	292.71	510.63	803.34
QTD	374.28	87.90	106.96	18.29	0.00	587.43	1240.91	1828.34
YTD	374.28	87.90	106.96	18.29	0.00	587.43	1240.91	1828.34

EMP NO. 11243 NAME: RHETT BUTLER — SS # 345-66-7777 — STATUS S — DATE/EMP 09/12/76 — PAY RATE 375.0000
ADDR: C/O GENERAL DELIVERY — EMPLOYEE TYPE 0 — FED-EX 1 — CHECK NO. 934 — H&W RATE 0.0000
TARA, CALIFORNIA — EMP. CLASS 0 — ST-EX 1 — CHECK DATE 02/12/77 — MON HOURS 80.5000
90009 — INS. CLASS 2 — ADD-EX 0 — VAC HOURS 70.00 — MON PAY 378.5200

	REG HRS	REG PAY	O.T. HRS	O.T. PAY	P.W. HRS	P.W. PAY	H&W PAY	VAC HRS	VAC PAY	OTHER PAY	NON-TAX
CURRENT	80.00	375.00	0.50	3.52	0.00	0.00	0.00	0.00	0.00	0.00	0.00
QTD	80.00	1125.00	2.50	16.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00
YTD	80.00	1125.00	2.50	16.50	0.00	0.00	0.00	0.00	0.00	0.00	0.00

	FED W/H	ST. W/H	F.I.C.A.	S.D.I.	OTHER DED	TOTAL DED	NET PAY	TOTAL PAY
CURRENT	51.97	9.83	22.14	3.79	0.00	87.73	290.79	378.52
QTD	169.09	27.69	66.78	11.41	50.00	324.97	816.53	1141.50
YTD	169.09	27.69	66.78	11.41	50.00	324.97	816.53	1141.50

IMPRESSIVE PRODUCTS — DATE 02/22/78
PAYROLL JOURNAL 02/14/78 - 01/28/78 — PAGE 2

COMPANY TOTALS: — VAC HOURS 150.00 — MON HRS 336.25 — MON PAY 3873.11

	REG HRS	REG PAY	O.T. HRS	O.T. PAY	P.W. HRS	P.W. PAY	H&W PAY	VAC HRS	VAC PAY	OTHER PAY	NON-TAX
CURRENT	282.00	3037.50	10.25	85.61	4.00	150.00	0.00	40.00	0.00	600.00	0.00
QTD	442.00	9112.50	17.25	136.09	4.00	150.00	0.00	40.00	0.00	625.00	0.00
YTD	442.00	9112.50	17.25	136.09	4.00	150.00	0.00	40.00	0.00	625.00	0.00

	FED W/H	ST. W/H	F.I.C.A.	S.D.I.	OTHER DED	TOTAL DED	NET PAY	TOTAL PAY
CURRENT	1001.20	264.83	226.58	38.73	100.00	1631.34	2241.77	3873.11
QTD	2291.28	553.97	580.58	99.23	150.00	3675.06	6348.53	10023.59
YTD	2291.28	553.97	580.58	99.23	150.00	3675.06	6348.53	10023.59

The payroll journal for Impressive Products provides current, quarter-to-date and year-to-date data in 19 categories plus almost as many additional items of information, on an impressive group of employees.

business computing

Payroll: Radio Shack

Stephen B. Gray

The User Instruction Manual for the Radio Shack TRS-80 "Business Systems: Payroll" is a black three-ring binder containing four pages of instruction, five payroll worksheets, five "Data Tape Generation List" forms, five identical "File Documentation" sheets that give the order of the eight program variables on the data tape, and five identical "Employee Records" sheets that give the order of the 18 items on the tape. The back cover holds eight cassettes, one for the Checks Program, another for the Update and Summary Program, and the other six for payroll data.

The manual opens with these two paragraphs:

"Radio Shack Payroll is a complete computer and manual system designed to reduce the workload involved in writing paychecks and keeping account balances. It contains two program tapes and six data tapes. The programs are: PAYROLL CHECKS and PAYROLL UPDATE & QUARTERLY SUMMARY. The data tapes are blank originally but both of the programs will write information on them. Be sure to keep careful records of what is on each tape. Sound accounting procedures are even more important on a computer system than on a manual system.

"As a second precaution, duplicate the programs two or three times. The procedure is described in the User's Manual. This will assure of always having a program should a tape be accidentally destroyed or erased."

(As a sound precaution, Radio Shack has recorded the two programs four times on each cassette, twice per side.)

How To Set Up Your System

First, load the Update and Summary tape into the TRS-80 computer. When you then type RUN, the computer responds with:

```
TYPE 1 TO CREATE A PAYROLL DATA TAPE
TYPE 2 TO GET A QUARTERLY SUMMARY
TYPE 3 TO ADD NEW EMPLOYEES TO THE
DATA TAPE
```

If you wish to create a payroll data tape, typing 1 and pressing ENTER will bring up:

```
LOAD A NEW TAPE--PRESS RECORD
PRESS ENTER WHEN READY
```

So you remove the program cassette from the tape recorder, insert one of the six blank data tapes, press both RECORD *and* PLAY simultaneously, and then ENTER. The computer then asks a series of questions:

```
HOW MANY EMPLOYEES?
WHAT IS THE FICA SALARY LIMIT?
WHAT IS THE FICA PERCENT?
TYPE 1 IF YOU HAVE STATE INCOME TAX
TYPE 2 IF YOU DO NOT
```

If you type 1, then the computer asks:

```
TYPE 1 IF STATE TAX IS A STRAIGHT
PERCENTAGE
TYPE 2 IF YOU JUST CALCULATE THE TAX
YOURSELF
```

The manual says: "All but six states are straight percent of either gross or Federal. If you operate out of one of those six, you must calculate State tax yourself each time. If State tax is a straight percent, the computer will ask for that percent." If you now type 1, the computer will ask

```
WHAT IS THE PERCENTAGE?
```

and after you've entered the figure, the computer asks you to

```
TYPE 1 IF THAT IS A PERCENTAGE OF
GROSS
TYPE 2 IF THAT IS A PERCENTAGE OF
FEDERAL TAX
```

After you enter 1 or 2, the computer starts in on City tax:

```
TYPE 1 IF YOU HAVE CITY TAX
TYPE 2 IF YOU DO NOT
```

If you type 1, then the computer asks the same questions as for State tax, on whether the City tax is a straight percentage or if you must calculate it yourself, what is the percentage, and then:

```
TYPE 1 IF YOU USUALLY HAVE DEDUC-
TIONS FOR THE CATEGORY TITLED---
OTHER---
TYPE 2 IF YOU DO NOT
```

This is a category "for any deductions not otherwise provided for, such as parking, non-resident taxes, etc."

The computer then goes on with further questions:

```
HOW MANY PAY PERIODS PER YEAR?
```

The answer, of course, is 52, 26, or 24. Then, after you respond to:

```
HOW MANY DOLLARS DEDUCTIBLE PER
DEPENDENT?
```

the screen comes up with

```
WRITING TO TAPE
```

About five seconds pass before the next message comes up on the screen, so if you haven't put in a blank data tape beforehand, you'll have to start all over again. The next message is:

```
TYPE 1 IF YOU WANT TO ENTER YEAR-TO-
DATE AND QUARTER-TO-DATE TOTALS
TYPE 2 IF YOU WANT ZEROS IN THESE
FIELDS
```

Next, for each employee, these questions are asked:

```
EMPLOYEE #1
ENTER FIRST 5 DIGITS OF SOC SEC $ (NO
DASHES)
ENTER LAST 4 DIGITS
TYPE 1 FOR SINGLE--2 FOR MARRIED
NUMBER OF DEPENDENTS
ENTER 1 FOR SALARIED--2 FOR HOURLY
```

If the employee is salaried, the question is:

```
ENTER DOLLARS PER PAY PERIOD
```

If hourly:

```
ENTER DOLLARS PER HOUR
```

If you said you'd wanted year-to-date and quarter-to-date totals, the computer next asks for:

```
EMPLOYEE #1
QUARTER-TO-DATE
GROSS
FED
FICA
STATE
CITY
OTHER
```

if you have indicated that all these deductions will be made. The computer then asks for the year-to-date amounts in the same deduction categories, writes these last two sets of data to tape, and goes on to ask the same questions about employee #2. When you've input all the data on all the employees, the computer notes:

```
DATA TAPE CREATED--END OF PROGRAM
```

As the manual says, "you are now ready to run your payroll on the TRS-80."

On Payday

First you load into the TRS-80 that other program tape, which is the Checks program. Then you put into the tape recorder the Payroll Data Tape you've just created (or which you created last payday). Press PLAY on

the recorder so the computer can read data from the tape as required by the Checks program.

Just to make sure you put the data tape into the recorder, the computer prints:

```
PAYROLL CHECKS
LOAD TAPE
PRESS ENTER
```

And when you press ENTER, the computer comes up with:

```
NEW QUARTER--1=YES, 2=NO
```

If you reply 1, the quarter-to-date totals will be set to zero. The computer reads the first employee's data from the tape, prints it on the screen, and then asks you to

```
ENTER REG HRS, OVTM HRS
```

if the employee is on an hourly wage. The tax is then calculated and printed on the screen under the same column headings used by the quarter-to-date and year-to-date totals.

Just in case any changes have taken place since you last wrote the tapes, the computer prints out at the bottom of the screen:

```
ANY CHANGES??? 1=NO, 2=GROSS, 3=FED,
4=FICA, 5=STATE, 6=CITY, 7=OTHER
```

To change any parameter, type in the relevant code. For instance, if you want to change the Federal tax, type 3; the message

```
NEW VALUE
```

comes up, you type the new amount for Federal tax, and within a moment after you hit ENTER, the data on the screen is changed to reflect the new rate.

Once you're satisfied that all the amounts are correct, reply with a 1, which means "no more changes," the computer will calculate net pay, and will update the QTD and YTD totals on the screen.

The Paycheck

The message now at the bottom of the screen is:

```
HIT ENTER TO GO ON
```

and when you do, the computer clears the screen and writes the paycheck by printing the amounts for GROSS, FEC, FICA, STATE, CITY, OTHER, and NET PAY.

After the paycheck is written, press ENTER and the next employee's data will be read into the computer.

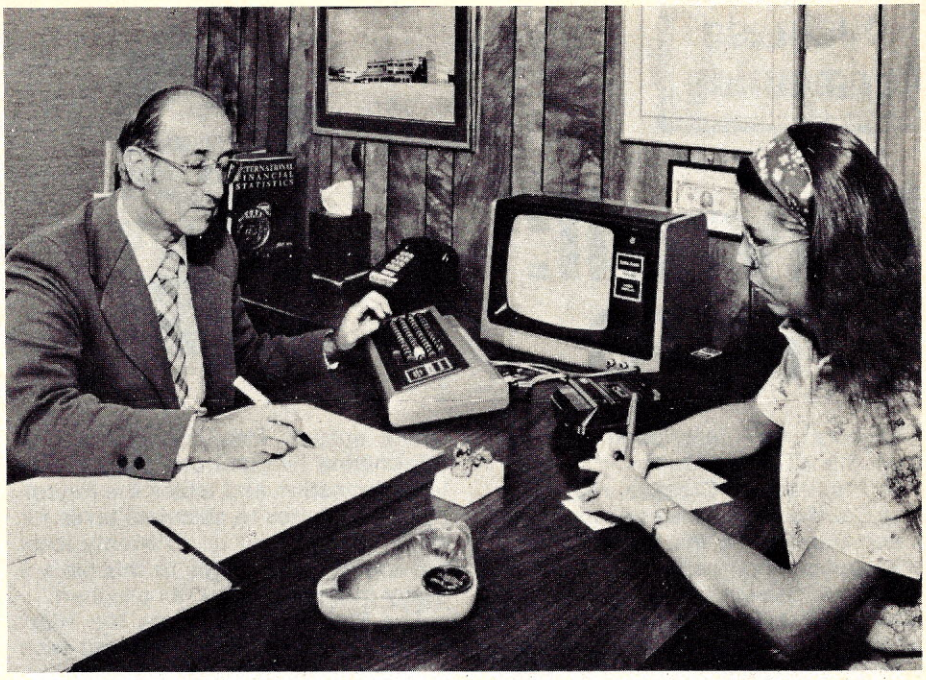
After all checks have been written, this message appears:

```
# OF NEW EMP
```

and if any new people were hired, you reply with the number added, and the program will process the new employees. The program will then print:

```
NEW TAPE--RECORD--HIT ENTER
```

which means the program is ready to write a new data tape that will be read into the *next* PAYROLL CHECKS program. Put in a new tape, press RECORD and PLAY on the tape recorder, and then press ENTER. After the new tape has been written, label and date it, write the generation number on the File Generation Worksheet, and you're all set until the next payday.



Modifications

The manual notes that the withholding tables are stored in certain lines, says that "when tax tables change, it will be necessary to modify these statements," and then shows just how to make these modifications.

The manual then goes on to say "This program will handle 11 employees in a 4K RAM machine, 66 employees in 8K, or 177 in 16K. If you have a 4K machine and wish to increase its capability to employees, you may do so by eliminating some of the messages the computer prints on the screen." Then the manual tells how to do this.

Correcting the Data Tape

As the manual puts it, "Occasionally it may be necessary to correct some of the numbers on the data tape," due to entry errors, an employee getting married, an increase in the number of an employee's dependents, etc. A 26-line program is given, to be used for making such corrections.

Quarterly Summary

Way back at the beginning, you had three choices, and we've just finished looking at all of the first choice, creating a payroll data tape. The second choice was "to get a quarterly summary." If you had typed a 2, the computer would ask if you want a summary for each employee, or if you just want a company summary.

If you press 1, for a summary on each employee, the computer will read the tape for each one and print out the figures; then each time you press ENTER, a new employee SOC SEC # will come up on the screen, along with the relevant payroll data.

After showing you the data for each employee, the computer will then provide a company summary (which you could have gotten all by itself earlier, by pressing a 2), headed:

```
SUMMARY FOR ALL EMPLOYEES
ACTIVE AND TERMINATED
```

with both QTD and YTD figures for GROSS, FED, FICA, STATE, CITY and OTHER.

Adding New Employees

The third choice back at the beginning was for adding new employees to the data tape. If you'd typed a 3, the computer would ask you to

```
LOAD MOST CURRENT DATA TAPE—
PRESS PLAY
PRESS ENTER WHEN READY
```

The computer would read the tape, and while doing so would put up on the screen

```
READING TAPE
```

At the end of the tape, the computer asks

```
HOW MANY NEW EMPLOYEES?
```

and then asks you to enter the person's SOC SEC #, marital status, etc. This data is then added to the data tape, after which this comes up on the screen:

```
DATA TAPE CREATED--END OF PROGRAM.
```

Availability

The Payroll Program is $19.95, and can be ordered from any Radio Shack store. ■

business computing

Payroll: MITS

The Payroll System is designed to automatically report back to the MITS General Ledger provided the user has a dual disk system.

[Ed note: the description is taken from the MITS brochure.

In the Mar-Apr Creative, *the Inventory Control package was described as being supplied by the Altair Software Distribution Company. Some names have been changed. What we're talking about now is the MITS Payroll System from the Microsystems Division of the Pertec Computer Corp.; the Altair name will be used for personal-computer hardware and software.]*

The Accounting System is comprised of four modules — general ledger, receivables, payables and payroll.

The Payroll package allows a company to prepare its periodic payroll for hourly, salaried, and commissioned employees while accumulating the necessary information for tax reporting. It generates the monthly, quarterly, and annual returns to be filed with local, state and federal governments. It also prepares employee W-2's and maintains an up-to-date information reference for each employee. The payroll package includes tables for federal withholding and FICA as well as withholding for all 50 states and up to 20 cities from precomputed or user-generated tables. The package will automatically produce payroll checks at the user's option.

General Description

The Payroll System keeps a record on each employee, storing such information as unit name and address, deductions and exemptions, SSN, pay type, pay period, and current month, quarter, and year-to-date totals for all earnings and deductions.

Complete Employee File maintenance: add, change, delete, and list capabilities are available.

Employees may be paid weekly, bi-weekly, semi-weekly, or monthly and any combination of these may be present at the same time. Employees may be paid as hourly employees, salaried employees, or draw-plus-commission employees and any mix of these types may be used at the same time.

The system also contains a Tax Information File that can store all the tax information and tables the Payroll System requires to calculate taxes for all fifty states and up to twenty local governments. Changes to this file are easily made using the tax program.

The Payroll System uses the information in both the Tax Files and each employer's record to calculate net pay, all state, local and federal taxes and up to three other deductions, and gross pay. This data is stored in the employee record and used by the system to print pay checks.

The system also generates:

- An end-of-month report showing unemployment liabilities and totals for each deduction and tax.
- A 941-A report giving all information needed to fill out the 941 and any state quarterly reports.
- W-2 forms for each employee.

For a typical dual-disk hardware configuration, up to 400 employees may be processed.

The Payroll System is designed to automatically report back to the MITS General Ledger provided the user has a dual-disk system.

```
                          THE HARRIS SUPPLY CO.
                             PAYROLL SYSTEM
                          EMPLOYEE LIST - INQUIRY
                                05/17/77


101/MPT  PAUL T. MANAGER            MARITAL ST.=M   DATE EMP. =04/23/65  STATUS=A
         1254 RITZ AVE. NE.         FED.EXEMPT.=3   DATE TERM.=//0
         DECATUR GA. 30032          ST. EXEMPT.=3   PAY PERIOD = S    PAY TYPE = S
         254-65-4346                CITY=0  ST.=10  PAY RATE =     $760.000
CURRENT:HOURS OT OTHER   DATE   CK.NO.  DED.: INSURANCE    MISC #1      MISC #2
        0    0    0   05/06/77  6139            $12.50       $1.50        $0.00

                          CURRENT        MONTH          QUARTER         YEAR
EARNINGS  -REGULAR     :    $760.00       $760.00       $2,280.00      $6,840.00
          -OVERTIME    :      $0.00         $0.00           $0.00          $0.00
          -OTHER HRS.  :      $0.00         $0.00           $0.00          $0.00
          -COMMISSIONS :      $0.00         $0.00           $0.00          $0.00
          -MISC.       :      $0.00         $0.00           $0.00          $0.00
DEDUCTIONS-FICA        :     $44.46        $44.46         $133.38        $400.14
          -FEDERAL     :     $21.79        $21.79          $65.37        $196.11
          -STATE       :     $24.27        $24.27          $72.81        $218.43
          -LOCAL       :      $0.00         $0.00           $0.00          $0.00
          -INSURANCE   :     $12.50        $12.50          $37.50        $112.50
          -MISC. #1    :      $1.50         $1.50           $4.50         $13.50
          -MISC. #2    :      $0.00         $0.00           $0.00          $0.00

101/SSC  SUZI C. SECRETARY          MARITAL ST.=S   DATE EMP. =02/12/69  STATUS=A
         45 W. MANCHESTER #32       FED.EXEMPT.=1   DATE TERM.=//0
         ATLANTA, GA. 30306         ST. EXEMPT.=1   PAY PERIOD = S    PAY TYPE = S
         254-87-6745                CITY=0  ST.=10  PAY RATE =     $350.000
CURRENT:HOURS OT OTHER   DATE   CK.NO.  DED.: INSURANCE    MISC #1      MISC #2
        0    0    0   05/06/77  6140            $4.00        $1.50        $0.00

                          CURRENT        MONTH          QUARTER         YEAR
EARNINGS  -REGULAR     :    $350.00       $350.00       $1,050.00      $3,150.00
          -OVERTIME    :      $0.00         $0.00           $0.00          $0.00
          -OTHER HRS.  :      $0.00         $0.00           $0.00          $0.00
          -COMMISSIONS :      $0.00         $0.00           $0.00          $0.00
          -MISC.       :      $0.00         $0.00           $0.00          $0.00
DEDUCTIONS-FICA        :     $20.48        $20.48          $61.44        $184.32
          -FEDERAL     :     $51.31        $51.31         $153.93        $461.79
          -STATE       :      $5.35         $5.35          $16.05         $48.15
          -LOCAL       :      $0.00         $0.00           $0.00          $0.00
          -INSURANCE   :      $4.00         $4.00          $12.00         $36.00
          -MISC. #1    :      $1.50         $1.50           $4.50         $13.50
          -MISC. #2    :      $0.00         $0.00           $0.00          $0.00
```

Up-to-date payroll records of two employees are provided by the Employee List.

Specifications

Minimum Machine Requirements. Altair 8800 series computer or equivalent, with 48K (49152) bytes of RAM, one floppy disk unit, and an input/output terminal with at least 80 characters output per line.

Recommended Machine Requirements. Altair 8800 series computer or equivalent, with 48K (49152) bytes of RAM, two floppy disk units, a video display unit (CRT) for data entry and editing, and a hardcopy printer for output reports and listings.

Operating Software. Altair Disk BASIC Language, Version 4.0. All applications programs (with the exception of several machine language subroutines) are written using this interpretive BASIC. NOTE: Altair Disk BASIC must be licensed separately from the accounting packages.

Documentation. A three-part users manual for each accounting package is provided, having sections titled General Information, Systems Guide, and Operators Guide. Subjects include:

- General System Overview
- Hardware/Software Matchup
- A Sample Company, with Reports and Listings
- Glossary of Terms and Definitions
- Logic Flow Diagram
- Program Narratives
- Installation and Startup Procedures
- Operator Instructions, with Sample Displays
- Handling of Exceptions and Error Conditions
- Other Miscellaneous Information

Warranty. Any programming defects reported will be corrected without charge for a period up to thirty-six months after commencement of license. Unauthorized customer and/or dealer tampering of software will void warranty.

License. Packages available for a one-time license fee arrangement through any of the Altair computer centers. OEM and Software House licenses available.

Installation and Training. One-time license fee normally includes on-site installation and training of customer's personnel, terms and conditions of which are determined by customer and dealer.

Software Notes. Software is supplied to the customer on a floppy diskette and, depending on customer-dealer arrangement, will be configured for that customer's hardware system. Each of the four accounting packages contains from 14 to 20 separate programs, including utility programs for systems generation, file and diskette backup, error recovery, and diskette testing. ■

```
                    THE HARRIS SUPPLY CO.
                       PAYROLL SYSTEM
                      PAYROLL REGISTER
                         05/17/77

 - - - - - - - - - - - - - - - - - - - -

101MPT   PAUL T. MANAGER              TYPE=S     RATE= 760.000

   ---HOURS---   --EARNINGS-  --------DEDUCTIONS--------  ---TOTALS---
   REG=   0.00   REG= 760.00  FIC=  44.46    INS=  12.50  EARN= 760.00
   OT =   0.00   OT =   0.00  FED=  21.79    MI1=   1.50  DEDU= 104.52
   OH =   0.00   OH =   0.00  STA=  24.27    MI2=   0.00
                 COM=   0.00  CIT=   0.00                 *NET= 655.48
                 MIS=   0.00                          CHECK NO.


101SSC   SUZI C. SECRETARY            TYPE=S     RATE= 350.000

   ---HOURS---   --EARNINGS-  --------DEDUCTIONS--------  ---TOTALS---
   REG=   0.00   REG= 350.00  FIC=  20.48    INS=   4.00  EARN= 350.00
   OT =   0.00   OT =   0.00  FED=  51.31    MI1=   1.50  DEDU=  82.64
   OH =   0.00   OH =   0.00  STA=   5.35    MI2=   0.00
                 COM=   0.00  CIT=   0.00                 *NET= 267.36
                 MIS=   0.00                          CHECK NO.


*** TOTALS - DEPARTMENT 101

 ---EARNINGS---  ----------DEDUCTIONS----------  ----TOTALS-----
 REG   1,110.00  FIC      64.94  INS      16.50  EARN   1,110.00
 OT        0.00  FED      73.10  MI1       3.00  DEDU     187.16
 OH        0.00  STA      29.62  MI2       0.00
 COM       0.00  CIT       0.00                  *NET     922.84
 MIS       0.00
```

The Payroll Register for Department 101 lists all the information required for calculating pay and writing paychecks.

For further information on the MITS Payroll System, contact the Microsystems Division, Pertec Computer Corp., 20630 Nordhoff Blvd., Chatsworth, CA 01311.

```
103SGT      THE HARRIS SUPPLY CO.      GEORGE T. SHIPPING          No. 06142
                 33 Northside Ave.
              Chamblee, Georgia 30340
--YOUR--  --HOURS WORKED--  ----------YOU EARNED----------
PAY RATE  REGULAR  PREMIUM   REGULAR    PREMIUM     OTHER
--------  -------  -------  --------  --------  --------      ***GROSS***
   3.950     40.0      0.0    158.00      0.00      0.00         158.00

---GOVERNMENT TAX DEDUCTIONS---  ---OTHER DEDUCTIONS---
  FICA  FEDERAL  STATE   LOCAL   INSUR.  MISC 1  MISC 2
-------  -------  -------  -------  ------  -------  -------      ****NET***
   9.24     6.60     0.78     0.00    8.00     0.00     2.50         130.88

----------------------YEAR TO DATE TOTALS----------------------
PAID THRU   EARNINGS     FICA      FEDERAL     STATE      LOCAL
---------  ---------  ---------  ---------  ---------  ---------
 05/06/77
            3,094.84     190.27     150.58      22.73       0.00
```

```
THE HARRIS SUPPLY CO.                  The Merchants National Bank      65 404/312      No. 06142
33 Northside Ave.                      Atlanta, Georgia
Chamblee, Georgia 30340
                                                                        Check No.- 6142

                          **** ONE HUNDRED THIRTY & 88 /100  DOLLARS
                                                          AMOUNT
                                  DATE
                                  05/05/77                ****$130.88
PAY TO THE ORDER OF:

   GEORGE T. SHIPPING
   88 MADDOX ST.
   ATLANTA, GA. 30329

                                                  NON NEGOTIABLE
```

PLACE MICR ENCODING HERE

The paycheck portion of the Payroll System provides both a check and a stub that shows all taxes, deductions and year-to-date totals.

business computing

This program is especially designed for small companies having a standardized work week and whose employees are paid by the hour rather than salaried.

Payroll:
Scientific Research Inst.

```
                              MINI-LEDGER

DEPT. NUMBER          : 51
FIRM NAME             :SCIENTIFIC RESEARCH
FOR WEEK ENDING       :8/5/77


ALL SALES

    SUNDAY    MONDAY   TUESDAY  WEDNESDAY  THURSDAY    FRIDAY  SATURDAY

      0.00    345.12    456.72     126.90    671.81    572.62      0.00

  THIS WEEK     2173.17  LAST WEEK        567.43   TO DATE      2740.60


PAYROLL
                                    HRS    GROSS   FICA  FEDRL  STATE    MISC     NET
EMP#  SOC.SEC#       NAME          WKD    WAGES    AMT   WITH   WITH     DED    WAGES

6732 237-62-1932 J. M.  DOYLE      65   424.45  24.83  55.28   0.00    8.93   335.41
3112 698-32-2679 G.R. CRAMER       80   254.40  14.88  33.02   0.00    7.50   199.00
918  183-67-2110 PHIL JOHNSON      80   350.40  20.49  42.86   0.00   16.50   270.55
A671 823-16-4312 J. SWAIN          80   460.00  26.91  54.69   0.00    8.50   369.90
                          TOTALS   80  1489.25  87.11 185.85   0.00   41.43  1174.86


  A/P

AC#   CHECK #     DATE       TO WHOM                    EXPLANATION        AMOUNT

 100    4567     8/2/77     LAFAYETTE                  MERCHANDISE         87.12
2400    4568     8/3/77     GEO SUPPLY                 OFFICE EXPEN        45.16
1500    4569     8/3/77     RADCO INC.                 SHOP EXPENSE        56.75
                                                                          ---------
OPENING BALANCE       123.45   CLOSING BALANCE      312.48   TOTAL        189.03


PETTY CASH

  DATE          TO  WHOM                    AMOUNT

8/2/77      ED PHILLIPS                      12.85
8/3/77      UPS                               8.63
                                            ------
OPENING BAL      25.13          TOTAL        21.48   CLOSING BAL      46.61
```

Among the summaries printed by the ledger program in Volume VI is this payroll update.

[*Ed. note: The following is taken mainly from SRI's manuals.*]

Scientific Research Inst. has three payroll programs. The first is in Volume III of their BASIC Software Library, "Advanced Business" ($39.95) by R.W. Brown, and can be input from audio cassette. There are no external files, because all the data is contained in DATA statements.

The second payroll program is in the front of Volume VI, "A Complete Business System" ($49.95, same author), but does not generate payroll checks. Instead, it is part of a ledger program that "performs periodic updates to the ledger files and also generates Payroll, Sales, A/P, Cash and Expense statistics. From these totals, the Balance Sheet, P&L, Year End taxes, 941's and W2 information may be generated." This ledger program is a module of a large system, which is a disk interactive version of (although not identical in all parts to) the Volume III program. The user can make up a business system from both Volumes III and VI, using disk interactive programs where desired.

The third program is described extensively in the back of Volume VI; and as the forward puts it, "the entire source code for this complete business system program is not included due to its proprietary matter." This proprietary package is available from Scientific Research Inst., 220 Knollwood, Key Biscayne, FL 33149.

First Payroll Program

The first payroll program, which takes up a little over seven pages and about 350 lines of BASIC statements, is described this way in Volume III:

Description

This program calculates and compiles a payroll register for all of your employees. All employee data is contained within the program so that external data files need not be used. The program computes the deductions for FICA and Federal and State income-tax withholdings, permits deductions for employee insurance, calculates the employer's state and federal unemployment insurance tax and has space provided for an additional deduction calculation (Union dues, emp. savings, loan repymts, etc.) should such space be needed or required. Four different printouts are generated by this program: (1) printing of paychecks, (2) Payroll Register; in a tabulated format, (3) employee data record and (4) a summarized tax record for the employer.

Users

This program is especially designed for small companies having a standard work week and whose employees are paid by the hour rather than salaried.

Instructions

Before the program is run, all employee data must initially be entered into the program. The program is well documented and should be listed for full details. The subroutines for the Federal taxes; line 1475, the Federal deduction schedule; line 1550, tax rate due; line 1585 and the State withholding; line 1680, should all be completed for your particular requirements before the program is initially run.

Limitations

This program is set for a maximum of 50 employees. This can be adjusted to accommodate other numbers of employees by changing the DIM statements in lines 155 and 160. The source code requires 9K bytes of memory for storage and 15K bytes of memory for execution, with 50 employees. A sample run of this program follows the source listing. The data generating the examples is contained within the program and should be removed before entering your data.

Payroll Modifications

There are only two modifications that may be made to enhance the operation of this program. They were not made before inclusion of the program in its present form because of the compatibility problems that exist between Advanced Basic compilers. The first and foremost modification would be to change the PRINT statements to PRINT USING statements in the report printouts. The second modification would be to use a Data File to feed information to the program instead of using internal DATA statements. If these modifications are not made, the payroll program will still function normally and without error. These changes are primarily intended as a convenience factor, not as a necessity.

The PRINT statements in all four of the reports may be converted to PRINT USING statements for added report clarity. The PRINT statements in the following lines should be converted: 835, 865, 890, 905, 1050, 1070, 1100, 1235, 1240, 1450 and 1460. The TAB( ) spacing for printing on the check and stub are listed on lines 630 to 680. By changing these tab constants, the spacing between fields on the check and stub may be altered to conform to the particular check form in use.

The program may also be modified to allow the entry of employee data from a Use File rather than from the Internal Data statements. This would require the addition of a FILES statement and conversion of the READ statements to READ # statements. The exact file modifications will depend on the computer system and Basic compiler version being used. If the program is modified for a Use File, delete the DATA statements in the program. For additional memory efficiency, the PRINT and PRINT USING statements should be merged with the READ # statements. This merger should not be made unless the program has been

```
THIS IS THE PAYROLL SECTION.  IT GENERATES THE FOLLOWING REPORTS.

              1 - PAYROLL LEDGER
              2 - PRINT PAYCHECKS OR 941'S
              3 - PRINT THE UNEMPLOYMENT TOTALS
              4 - WITHHOLDING TOTALS
              5 - UPDATE EMPLOYEE DATA
              6 - END

WHICH ONE DO YOU WANT TO DO? 1



PAYROLL LEDGER                                        DATE JUL. 23, 1977

EMPLOYEE      EMPLOYEE        S.S.          $/HR.       #
NUMBER        NAME            #                         DEPENDENTS

MISC.         GROSS           TOTAL         TOTAL       TOTAL          INS.
DED           PAY             FED           STATE       FICA 11        DED.

---------------------------------------------------------------------------

A3721         J.M. DOYLE      339-26-4096   $6.53             2
  $0.00         $7,169.94     $1,403.85     $426.37       $419.44        $7.50

B6219         G.R. CRAMER     224-16-3209   $3.73           103
 $10.00         $3,604.34     $608.62       $208.89       $210.84        $9.04

A1872         PHIL H. JOHNSON 118-19-3402   $4.38             1
  $0.00         $4,905.60     $1,047.85     $292.40       $286.96        $6.15

E4531         G.L. HARRISON   364-45-8237   $3.90             2
  $0.00           $0.00       $0.00         $0.00         $0.00          $0.00
```

In the third SRI payroll program, the user has selected item 1 from the menu, the payroll ledger. (The printout of the *first* payroll program is simpler: it shows name, employee number, net pay, total tax, total deductions, and the totals of these last three figures.)

modified for a Use File.

Third Payroll Program

Much of the explanatory text accompanying the programs in the latter two-thirds of Volume VI, called "A Complete Business System, ACBS rev:80," was reprinted in the SRI Inventory Control article in the March-April issue of *Creative Computing* (pages 116-120). One of the paragraphs bears repeating:

Payroll

Yearly changes or updates to the tax algorithms in the payroll section are accomplished by typing in the number of the line or lines; one at a time, followed with the amended line data. Federal taxes are for the year 1977 and State taxes have been set to that used by the state of Maryland for 1977. After the changes have been made, save them by typing SAVE "PAY PROG. This will delete the old disk copy and replace it with the amended program. Do not try to run this except in the normal manner by typing RUN" ACBS otherwise the ACBS programs may be damaged or destroyed.

* * * * * * *

The end of Volume VI contains a section on Yearly Payroll Tax Updating. The text says:

The following sheet is a listing of the employee tax algorithms. To change the State or Federal tax withholdings simply type LOAD"PAY PROG. Then type in the line number you wish to change followed by the entire line as shown on the following page, substituting the numbers you are changing too, in place of the numbers that appear on the line. If you need to delete a line, simply type the line number and then press the Return key. If you need to enter a new line in addition to what is already there, such as in the State tax section, simply type in a line number between the two lines where you wish to place the new line and then type in the new line information. When finished, press the Return key. When all of the updating you are going to do is done, type SAVE"PAY PROG. You now have saved an updated copy of the payroll program. ■

```
TYPE A '0' IF YOU WANT TO RUN THE 941 QUARTERLY TOTALS.
 OR TYPE A '1'IF YOU WANT TO RUN PAYCHECKS.? 1

DID ALL THE EMPLOYEES WORK A STANDARD PAY PERIOD (Y OR N)? Y

WHEN CHECKS ARE IN PLACE, READY FOR PRINTING, TYPE IN THE STARTING CHECK #? 479
```

Scientific Research Instruments Co., Inc.
PAYROLL ACCOUNT
1712 Farmington Court, Crofton, Md. 21114

15-3
511

No. 479

DATE JUL. 23 , 1977

PAY TO THE ORDER OF: J.M. DOYLE

PAY
$ 374. 44

PAY EXACTLY $374. DOLLARS AND 44 CENTS

Scientific Research Instruments Co., Inc.
PAYROLL ACCOUNT

MAIN OFFICE
THE RIGGS NATIONAL BANK
WASHINGTON, D.C.

AUTHORIZED SIGNATURE

THIS IS A STATEMENT OF YOUR EARNINGS AND DEDUCTIONS
PLEASE DETACH AND RETAIN

No. 479

PERIOD ENDING JUL. 23, 1977

		DEDUCTIONS					
A3721	522.40		7.50			80	
EMP. NO.	SALARY	INSURANCE			SAVINGS		DEPT. NO.

	$7,169.94		0.00	80.22	30.56	29.68	$374.44
OVERTIME OR COMMISSION	GROSS SALARY	ADVANCES	OTHER DEDUCTIONS	FED. W.H. TAX	STATE W.H. TAX	F.I.C.A.	NET PAY

Scientific Research Instruments Co., Inc.

FILE STRUCTURES

[The first five file structures, given in detail in the March-April *Creative Computing*, were for files named MISC, A/P, A/R, EINV, and MINV.]

File #6 - PAY

# of employees, # of hours in each pay period, State Unemployment %, employee #, social security #, Active Emp., Name, St. Add, City, State, Zip code, Rate of pay $, # Dependents, $ Deductions (Ins. etc.), Misc deductions $, total $ Gross pay (year to date), total $ FICA withheld year to date, total $ Federal Tax withheld year to date, total $ State Tax held year to date, total $ Gross pay this period to date (or since the last P & L), total $ Taxes paid for employee this period (i.e.: U.I., State taxes, FICA, etc.) Wages this quarter, FICA this quarter, Fed. taxes this quarter.

Selecting item 2 on the menu prints paychecks, with a user-selected starting check number. (The address is SRI's previous location.)

..short programs...

Systematic Savings Revisited

Stu Denenberg

Referring to "Systematic Savings" on page 132 of the Nov/Dec '77 *Creative Computing*, the fancy mathematics formula masks what is happening.

Why not just do the calculation as a person would do with a hand calculator? We could begin with the simpler problem of calculating compound interest and then slightly modify that procedure to do systematic investments. For example, the Basic program for compound interest is:

```
5 PRINT "AT END OF YEAR","BALANCE"
10 READ P,R,N
20 FOR I=1 TO N
30 P=P+P*R
40 PRINT I,,P
50 NEXT I
60 STOP
70 DATA 100,.1,10
80 END
```

```
RUN
AT END OF YEAR          BALANCE
 1                       110
 2                       121
 3                       133.1
 4                       146.41
 5                       161.051
 6                       177.156
 7                       194.872
 8                       214.359
 9                       235.795
 10                      259.374
```

Note especially that Line 30 is *not* P= P* (I + R); instead it stresses what we *actually* do when we calculate interest — namely multiply the principal by the interest rate and then add that back onto the principal to give the new principal.

Now the program to do systematic savings is exactly the same as the one for compound interest but instead of letting our 100 bucks lay around all lonely while it's compounding, we keep feeding in lumps of $100 at the end of each year so now the program looks like:

```
5 PRINT "AT END OF YEAR     AMOUNT INVESTED     TOTAL ACCUMULATED"
10 READ N,C,R
20 P=C
30 FOR I=1 TO N
40 P=P+P*R
50 PRINT TAB(5);I;TAB(25);I*C;TAB(45);P
60 P=P+C
70 NEXT I
80 STOP
90 DATA 10,100,.1
100 END
```

```
RUN
AT END OF YEAR      AMOUNT INVESTED      TOTAL ACCUMULATED
      1                  100                  110
      2                  200                  231
      3                  300                  364.1
      4                  400                  510.51
      5                  500                  671.561
      6                  600                  848.717
      7                  700                  1043.59
      8                  800                  1257.95
      9                  900                  1493.74
      10                 1000                 1753.12
```

C is the constant amount we save each year. Line 60 is the only *real* difference between the two programs and it shows how we add in the constant savings to our principal each year.

Dr. Stuart Denenberg
Dept. of Computer Science
SUNY
Plattsburgh, NY 12901

Compound Interest

If $1000 is deposited in a savings account paying 8%. interest compounded n times a year, then this will accumulate to

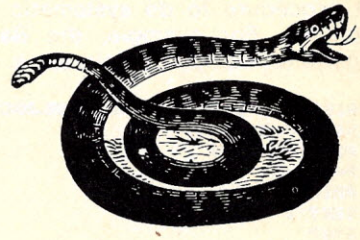
$$\$1000(1 + .08/n)^{n}$$

at the end of one year assuming that no deposits or withdrawals are made.

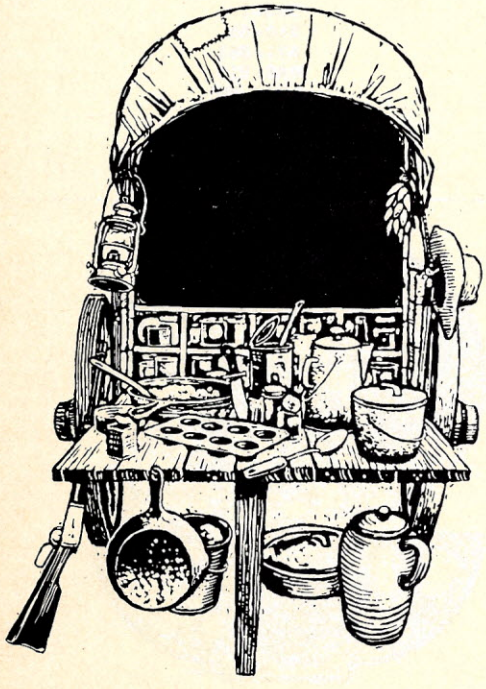
n	8% Compounded	Accumulation at end of one year. (Rounded to nearest cent)	
1	Yearly	$\$1000(1+.08/1)^{1} =$	$1080.00
2	Semiannually	$\$1000(1+.08/2)^{2} =$	$1081.60
4	Quarterly	$\$1000(1+.08/4)^{4} =$	$1082.43
12	Monthly	$\$1000(1+.08/12)^{12} =$	$1083.00
365	Daily	$\$1000(1+.08/365)^{365} =$	$1083.28
8760	Hourly	$\$1000(1+.08/8760)^{8760} =$	$1083.29
525,600	Every minute	$\$1000(1+.08/525600)^{525600} =$	$1083.29
31,536,000	Every second	$\$1000(1+.08/31536000)^{31536000} =$	$1083.29

Hardly worth quibbling over hours, minutes, and seconds.

OREGON TRAIL



Dan Rawitsch



This program simulates a trip over the Oregon Trail from Independence, Missouri to Oregon City, Oregon in 1847. Your family of five will cover the 2040-mile Oregon Trail in 5-6 months — if you make it alive.

INTRODUCTION

> "The Santa Fe Trail being first established, a signboard was later set up to show where the Oregon Trail branched off. It bore the simple legend 'Road To Oregon.' ... Surely so unostentatious a sign never before nor since announced so long a journey."
>
> - H.M. Chittenden
> *The American Fur Trade of the Far West*

During the thirty-year period from 1840 to 1870, thousands of pioneers traveled over the 2000-mile Oregon Trail tc settle on the West Coast. The history of the trail may be seen by some as a strong example of heroic American themes such as "conquering the frontier" and "the pioneer spirit." To others, the great western migration carries the political overtones of the colonists and their descendents forcing out British imperialism and clearing away the native American Indians in an effort to dominate middle North America themselves. At the very least, the journey over the trail represents the human stories of many individuals who, oblivious to historical trends, tried to survive in life as best they knew how.

RATIONALE FOR COMPUTER USAGE

This computer simulation, developed initially in 1971 and refined in 1975, is an attempt to give students a better feeling of what the journey west was like for the people who attempted it. Like all simulations, OREGON does not attempt to replicate exactly a trip on a wagon train in the 1840's. But it does attempt to present students with some of the resources, decisions, and events that faced the pioneers of that day. Although students can find out about the Oregon Trail by reading books, visiting museums, watching movies, and similar activities, the simulation allows them to learn from actively participating in the simulated experiences of people from another era.

Background On The OREGON Program

In 1971, Don Rawitsch and Bill Heinemann were participating together in a practice teaching program as students at Carleton College, Northfield, Minnesota. Don was teaching a class on the history of the American West and provided the preliminary information which Bill, a math teacher, used to construct the OREGON program. The program was first implemented on the Minneapolis Schools timesharing system. On the completion of the practice teaching program, the program was removed from the Minneapolis system and remained only as a curled up listing until Don joined the MECC staff in 1974 and loaded it onto the MECC system. Don then proceeded to do further research on the Oregon Trail and modified the program for historical accuracy to produce the present version. The program has been implemented on Hewlett-Packard, UNIVAC, and Control Data systems.

HISTORICAL BACKUP INFORMATION

Although historical information about the trip to Oregon is not extremely plentiful, primary and secondary sources were used whenever possible to make the simulation authentic.

- Mileage and route of the Trail

  Morgan, insert-back cover, "Map of T.H. Jefferson - 1849"

  Hancock, xiv, xv, information based on a map of 1846

  Meeker, *Ox Team Days,* pp. 252-53

  Meeker, *Ox Team Days,* p. 61 - says travelers averaged 15-25 miles per day, though they didn't travel every day. In the simulation, players make about 175-200 miles every two-week period.

  Ghent, p. 73 - says ox-drawn wagons made 2 miles per hour, or 20 miles on good days and 5-10 miles on bad days.

- Costs of resources

  Meeker, *Ox Team Days,* p. 13, says in 1850's sugar cost 18¢/lb., salt cost $3.00/barrel, calico cost 15¢/yd.

  Ghent, p. 99, says a team of oxen cost about $200 (for eight); references a guidebook of the time which recommends the following to be included for each adult:

  150 lbs. of flour — 25 lbs. of bacon
  25 lbs. of sugar — 15 lbs. of coffee

  In the simulation, the player spends $200-$300 on an oxen team. Based on the Meeker information, if the average commodity cost about 20¢/lb. and the average family of five eats as much as four adults, a good food stock would cost about $175. This is a reasonable amount to start with in the simulation.

- Frequency of misfortunes occurring

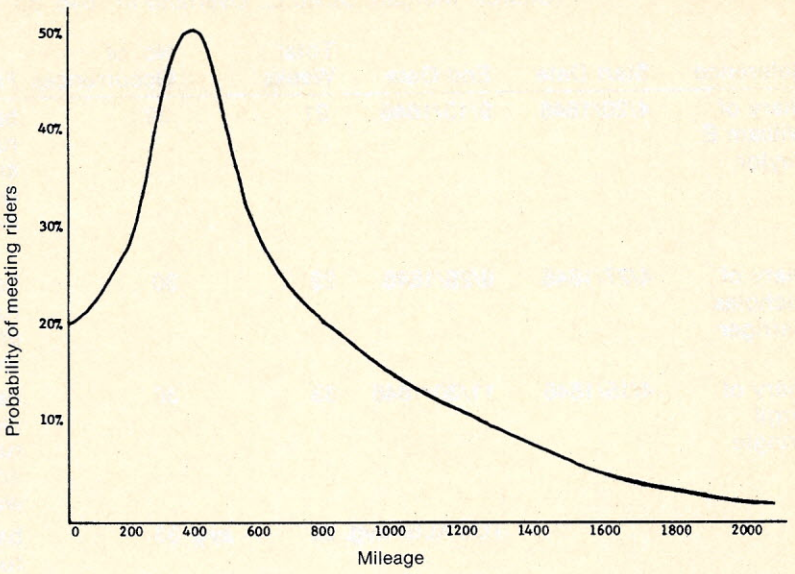
  Table 1 shows a frequency analysis of events mentioned in the diaries of three people that traveled the entire length of the trail. The probabilities of events occurring in the simulation are based on this analysis.

- Miscellaneous

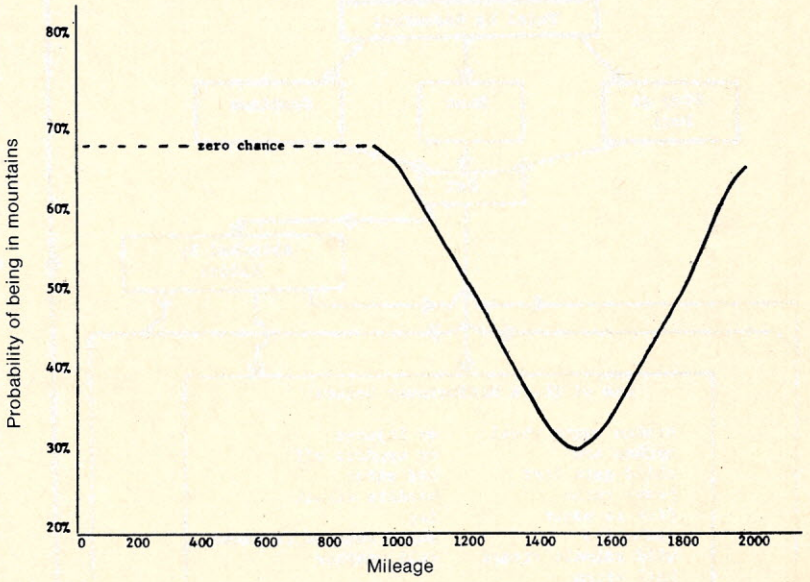
  Dates and days of the week shown are correct for 1847.

  The average trip in the simulation takes about 12 two-week turns. There were six forts on the trail. In the simulation a player gets the option to stop at a fort every other turn.

Probability curves for being attacked by riders and for being in the mountains are representative of the geographic features of the land. (Riders attack more frequently on the plains.)



Occurrence of "Riders Ahead" as a function of mileage



Occurrence of "Rugged Mountains" as a function of mileage

BIBLIOGRAPHY

Ghent, William J., *The Road to Oregon,* Longmans, Green & Co., New York, 1929

Hancock, Samuel, *Narrative of Samuel Hancock,* George H. Harrap & Co., Ltd., London, 1927.

Meeker, Ezra, *Ox Team Days on the Oregon Trail,* pub. by E. Meeker, New York, 1907.

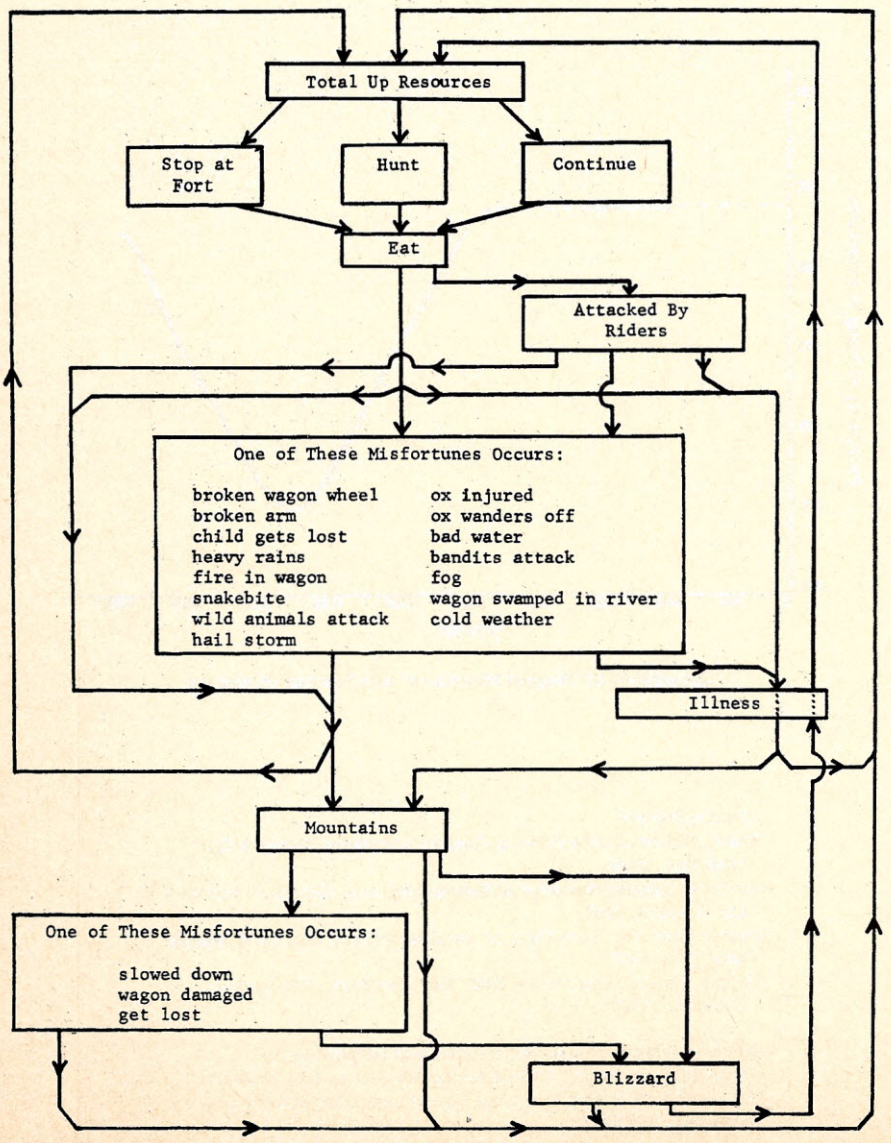
Morgan, Dale L., *Overland in 1846,* Talisman Press, Georgetown, California, 1963.

Extensive additional material; sources, and background are contained in the MECC OREGON User Manual by Don Rawitsch. Minnesota Educational Computer Consortium, 2520 Broadway Drive, Lauderdale, MN 55113.

INFORMATION FROM DIARIES OF PEOPLE TRAVELING THE OREGON TRAIL
(Source: Morgan, David L., *Overland in 1846,* pp. 123-130,150-158,163-188)

Reference	Start Date	End Date	Total Weeks	No. of Occurrences	Frequency of Occurrences			
Diary of William E. Taylor	4/20/1846	9/13/1846	21	32	bad weather	10	illness	4
					rugged trail	4	no water	3
					animals lost	1	injury	0
					wagon breakdown	2	bad river	1
					met friendly Indians	5	thieves	2
Diary of Nicholas Carriger	4/27/1846	9/26/1846	22	30	bad weather	2	illness	10
					rugged trail	6	injury	3
					animals lost	6	thieves	1
					wagon breakdown	2		
Diary of Virgil Pringle	4/15/1846	11/30/1846	33	38	bad weather	7	no water	5
					illness	5	injury	2
					rugged trail	7	bad river	3
					animals lost	3	thieves	2
					wagon breakdown	4		
		TOTALS: avg. 25		avg. 33	bad weather	20%	illness	20%
					rugged trail	16%	no water	8%
					animals lost	10%	injury	5%
					wagon breakdown	8%	bad river	4%
					met friendly Indians	5%	thieves	4%

Table 1

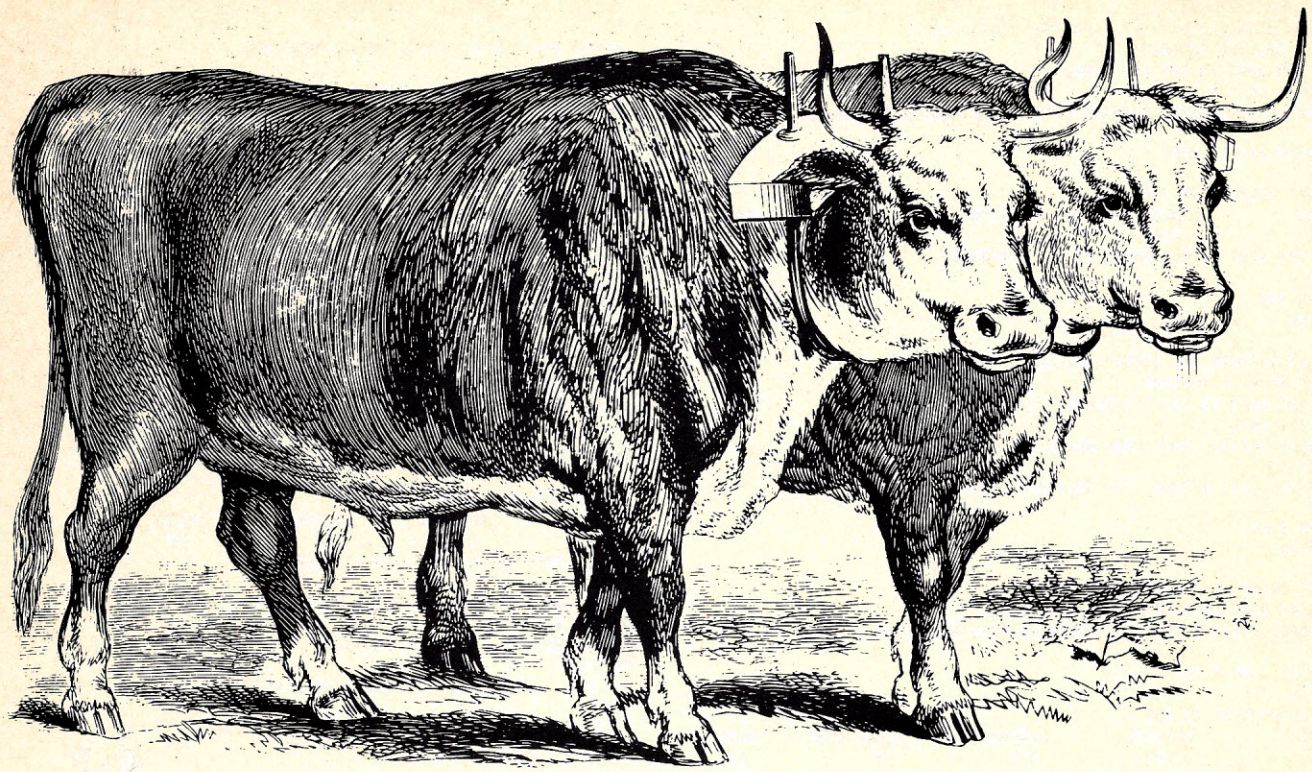


OREGON: Detailed Model

Background on MECC

The Minnesota Educational Computing Consortium (MECC) was created in 1972 out of concern by the governor and legislature that educational computing needed a central source of coordination for planning, and a mechanism to insure that all educational institutions in the state would have equal opportunity of access to computing services for both instructional and administrative programs. The Consortium's membership includes the University of Minnesota (5 campuses), The Minnesota State University System (7 campuses), the Minnesota Community College System (18 campuses), the Minnesota Department of Education (representing the state's 436 independent school districts), and the Minnesota Department of Administration. Minnesota is the only state in the country having a central organization for coordinating educational computing activities across all levels of education.

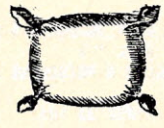
The MECC Instructional Services Division offers a variety of services to consortium members. A technical staff operates the largest of Minnesota's computers dedicated to instructional computing, a Control Data CYBER 73 time-sharing system. The MECC Timeshare System is currently configured for 375 user ports and serves about 1100 interactive terminals located in schools and colleges across the state. A large multiplexing communications network provides the route by which MECC users access the Timeshare System, whether they are a few miles from the Minneapolis-St. Paul computer center or hundreds of miles away near the Canadian border. The MECC User Services staff of instructional coordinators helps user learn to make better use of the computer by visiting school and college sites, conducting workshops, providing over-the-phone consulting service, publishing news letters, and producing written documentation for programs in the MECC Timeshare System's central library.



Sample Run

```
THIS PROGRAM SIMULATES A TRIP OVER THE OREGON TRAIL FROM
INDEPENDENCE, MISSOURI TO OREGON CITY, OREGON IN 1847.
YOUR FAMILY OF FIVE WILL COVER THE 2040 MILE OREGON TRAIL
IN 5-6 MONTHS --- IF YOU MAKE IT ALIVE.

YOU HAD SAVED $900 TO SPEND FOR THE TRIP, AND YOU'VE JUST
   PAID $200 FOR A WAGON.
YOU WILL NEED TO SPEND THE REST OF YOUR MONEY ON THE
   FOLLOWING ITEMS

     OXEN - YOU CAN SPEND $200-$300 ON YOUR TEAM
            THE MORE YOU SPEND, THE FASTER YOU'LL GO
               BECAUSE YOU'LL HAVE BETTER ANIMALS

     FOOD - THE MORE YOU HAVE, THE LESS CHANCE THERE
               IS OF GETTING SICK

     AMMUNITION - $1 BUYS A BELT OF 50 BULLETS
            YOU WILL NEED BULLETS FOR ATTACKS BY ANIMALS
               AND BANDITS, AND FOR HUNTING FOOD

     CLOTHING - THIS IS ESPECIALLY IMPORTANT FOR THE COLD
               WEATHER YOU WILL ENCOUNTER WHEN CROSSING
               THE MOUNTAINS

     MISCELLANEOUS SUPPLIES - THIS INCLUDES MEDICINE AND
               OTHER THINGS YOU WILL NEED FOR SICKNESS
               AND EMERGENCY REPAIRS


YOU CAN SPEND ALL YOUR MONEY BEFORE YOU START YOUR TRIP -
OR YOU CAN SAVE SOME OF YOUR CASH TO SPEND AT FORTS ALONG
THE WAY WHEN YOU RUN LOW.  HOWEVER, ITEMS COST MORE AT
THE FORTS.  YOU CAN ALSO GO HUNTING ALONG THE WAY TO GET
MORE FOOD.
WHENEVER YOU HAVE TO USE YOUR TRUSTY RIFLE ALONG THE WAY,
YOU WILL BE TOLD TO TYPE IN A WORD (ONE THAT SOUNDS LIKE A
GUN SHOT).  THE FASTER YOU TYPE IN THAT WORD AND HIT THE
"RETURN" KEY, THE BETTER LUCK YOU'LL HAVE WITH YOUR GUN.

AT EACH TURN, ALL ITEMS ARE SHOWN IN DOLLAR AMOUNTS
EXCEPT BULLETS
WHEN ASKED TO ENTER MONEY AMOUNTS, DON'T USE A "$".

GOOD LUCK!!!

HOW GOOD A SHOT ARE YOU WITH YOUR RIFLE?
  (1) ACE MARKSMAN,  (2) GOOD SHOT,  (3) FAIR TO MIDDLIN'
        (4) NEED MORE PRACTICE,  (5) SHAKY KNEES
ENTER ONE OF THE ABOVE -- THE BETTER YOU CLAIM YOU ARE, THE
FASTER YOU'LL HAVE TO BE WITH YOUR GUN TO BE SUCCESSFUL.
? 4

HOW MUCH DO YOU WANT TO SPEND ON YOUR OXEN TEAM ? 250
HOW MUCH DO YOU WANT TO SPEND ON FOOD ? 150
HOW MUCH DO YOU WANT TO SPEND ON AMMUNITION ? 50
HOW MUCH DO YOU WANT TO SPEND ON CLOTHING ? 150
HOW MUCH DO YOU WANT TO SPEND ON MISCELLANEOUS SUPPLIES ? 50
AFTER ALL YOUR PURCHASES, YOU NOW HAVE  50  DOLLARS LEFT


MONDAY MARCH 29 1847

TOTAL MILEAGE IS 0
FOOD            BULLETS         CLOTHING        MISC. SUPP.     CASH
 150             2500            150             50              50
DO YOU WANT TO (1) HUNT, OR (2) CONTINUE
? 2
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 1
THERE WAS A FIRE IN YOUR WAGON--FOOD AND SUPPLIES DAMAGED

MONDAY APRIL 12 1847

TOTAL MILEAGE IS 200
FOOD            BULLETS         CLOTHING        MISC. SUPP.     CASH
 97              2100            150             40              50
DO YOU WANT TO (1) STOP AT THE NEXT FORT, (2) HUNT, OR (3) CONTINUE
? 3
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 1
YOU KILLED A POISONOUS SNAKE AFTER IT BIT YOU

MONDAY APRIL 26 1847

TOTAL MILEAGE IS 409
FOOD            BULLETS         CLOTHING        MISC. SUPP.     CASH
 84              2090            150             35              50
DO YOU WANT TO (1) HUNT, OR (2) CONTINUE
? 2
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2
WAGON GETS SWAMPED FORDING RIVER--LOSE FOOD AND CLOTHES

MONDAY MAY 10 1847

TOTAL MILEAGE IS 580
FOOD            BULLETS         CLOTHING        MISC. SUPP.     CASH
 36              2090            130             35              50
DO YOU WANT TO (1) STOP AT THE NEXT FORT, (2) HUNT, OR (3) CONTINUE
? 2
TYPE WHAM
? WHAM
```

```
RIGHT BETWEEN THE EYES---YOU GOT A BIG ONE!!!!
FULL BELLIES TONIGHT!
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 3
WAGON GETS SWAMPED FORDING RIVER--LOSE FOOD AND CLOTHES

MONDAY MAY 24 1847

TOTAL MILEAGE IS 719
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 36            2078           110            35             50
DO YOU WANT TO (1) HUNT, OR (2) CONTINUE
? 1
TYPE POW
? POW

RIGHT BETWEEN THE EYES---YOU GOT A BIG ONE!!!!
FULL BELLIES TONIGHT!
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2
RIDERS AHEAD.  THEY LOOK HOSTILE
TACTICS
(1) RUN  (2) ATTACK  (3) CONTINUE  (4) CIRCLE WAGONS
? 2
TYPE BLAM
? BLAM

NICE SHOOTING---YOU DROVE THEM OFF
RIDERS WERE HOSTILE--CHECK FOR LOSSES
HELPFUL INDIANS SHOW YOU WHERE TO FIND MORE FOOD

MONDAY JUNE 7 1847

TOTAL MILEAGE IS 883
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 86            1984           110            35             50
DO YOU WANT TO (1) STOP AT THE NEXT FORT, (2) HUNT, OR (3) CONTINUE
? 3
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 1
WAGON GETS SWAMPED FORDING RIVER--LOSE FOOD AND CLOTHES
RUGGED MOUNTAINS
THE GOING GETS SLOW
BLIZZARD IN MOUNTAIN PASS--TIME AND SUPPLIES LOST

MONDAY JUNE 21 1847

TOTAL MILEAGE IS 950
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 18            1684           90             25             50
DO YOU WANT TO (1) HUNT, OR (2) CONTINUE
? 1
TYPE BLAM
? BLAM

RIGHT BETWEEN THE EYES---YOU GOT A BIG ONE!!!!
FULL BELLIES TONIGHT!
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2
RIDERS AHEAD.  THEY LOOK HOSTILE
TACTICS
(1) RUN  (2) ATTACK  (3) CONTINUE  (4) CIRCLE WAGONS
? 2
TYPE BLAM
? BLAM

NICE SHOOTING---YOU DROVE THEM OFF
RIDERS WERE HOSTILE--CHECK FOR LOSSES
HAIL STORM---SUPPLIES DAMAGED
RUGGED MOUNTAINS
THE GOING GETS SLOW
```



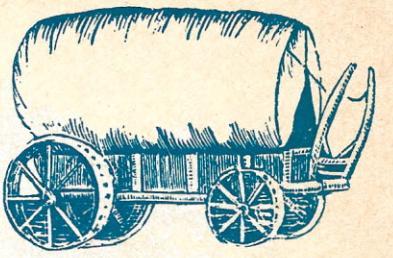
```
MONDAY JULY 5 1847

TOTAL MILEAGE IS 1004
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 53            1392           90             19             50
DO YOU WANT TO (1) STOP AT THE NEXT FORT, (2) HUNT, OR (3) CONTINUE
? 3
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2
WILD ANIMALS ATTACK!
TYPE POW
? POW

NICE SHOOTIN' PARDNER---THEY DIDN'T GET MUCH
MONDAY JULY 19 1847

TOTAL MILEAGE IS 1217
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 35            1392           90             19             50
DO YOU WANT TO (1) HUNT, OR (2) CONTINUE
? 2
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2

MONDAY AUGUST 2 1847

TOTAL MILEAGE IS 1429
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 17            1392           90             19             50
DO YOU WANT TO (1) STOP AT THE NEXT FORT, (2) HUNT, OR (3) CONTINUE
? 2
TYPE BANG
? BANG

RIGHT BETWEEN THE EYES---YOU GOT A BIG ONE!!!!
FULL BELLIES TONIGHT!
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2

MONDAY AUGUST 16 1847

TOTAL MILEAGE IS 1594
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 55            1380           90             19             50
DO YOU WANT TO (1) HUNT, OR (2) CONTINUE
? 2
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2
RIDERS AHEAD.  THEY LOOK HOSTILE
TACTICS
(1) RUN  (2) ATTACK  (3) CONTINUE  (4) CIRCLE WAGONS
? 2
TYPE WHAM
? WHAM

NICE SHOOTING---YOU DROVE THEM OFF
RIDERS WERE HOSTILE--CHECK FOR LOSSES
WILD ANIMALS ATTACK!
TYPE BLAM
? BLAM

NICE SHOOTIN' PARDNER---THEY DIDN'T GET MUCH
RUGGED MOUNTAINS
THE GOING GETS SLOW
BLIZZARD IN MOUNTAIN PASS--TIME AND SUPPLIES LOST
```

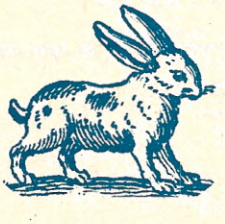
```
MONDAY AUGUST 31 1847

YOU'D BETTER DO SOME HUNTING OR BUY FOOD AND SOON!!!!
TOTAL MILEAGE IS 1685
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 11            999            89             9              50
DO YOU WANT TO (1) STOP AT THE NEXT FORT, (2) HUNT, OR (3) CONTINUE
? 2
TYPE POW
? POW

RIGHT BETWEEN THE EYES---YOU GOT A BIG ONE!!!!
FULL BELLIES TONIGHT!
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 2

MONDAY SEPTEMBER 13 1847

TOTAL MILEAGE IS 1847
FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 48            986            89             9              50
DO YOU WANT TO (1) HUNT, OR (2) CONTINUE
? 2
DO YOU WANT TO EAT (1) POORLY  (2) MODERATELY
OR (3) WELL ? 1
COLD WEATHER---BRRRRRRR!---YOU HAVE ENOUGH CLOTHING TO KEEP YOU WARM

YOU FINALLY ARRIVED AT OREGON CITY
AFTER 2040 LONG MILES---HOORAY!!!!!
A REAL PIONEER!

SATURDAY SEPTEMBER  25  1847

FOOD          BULLETS        CLOTHING       MISC. SUPP.    CASH
 35            986            89             9              50

          PRESIDENT JAMES K. POLK SENDS YOU HIS
               HEARTIEST CONGRATULATIONS

          AND WISHES YOU A PROSPEROUS LIFE AHEAD

                    AT YOUR NEW HOME

RUN COMPLETE.
```

Program Listing

```
10 REM PRØGRAM NAME - ØREGØN          VERSIØN:01/01/78
20 REM ØRIGINAL PRØGRAMMING BY BILL HEINEMANN - 1971
30 REM SUPPØRT RESEARCH AND MATERIALS BY DØN RAWITSCH,
40 REM      MINNESØTA EDUCATIØNAL CØMPUTING CØNSØRTIUM STAFF
50 REM CDC CYBER 70/73-26      BASIC 3.1
60 REM DØCUMENTATIØN BØØKLET 'ØREGØN' AVAILABLE FRØM
61 REM     MECC SUPPØRT SERVICES
62 REM     2520 BRØADWAY DRIVE
63 REM     ST. PAUL, MN  55113
80 REM
150 REM  *FØR THE MEANING ØF THE VARIABLES USED, LIST LINES 6470-6790*
155 REM
160 PRINT "DØ YØU NEED INSTRUCTIØNS  (YES/NØ)";
170 DIM C$(5)
180 REM RANDØMIZE REMØVED
190 INPUT C$
200 IF C$="NØ" THEN 690
210 PRINT
220 PRINT
230 REM ***INSTRUCTIØNS***
240 PRINT "THIS PRØGRAM SIMULATES A TRIP ØVER THE ØREGØN TRAIL FRØM"
250 PRINT "INDEPENDENCE, MISSØURI TØ ØREGØN CITY, ØREGØN IN 1847."
260 PRINT "YØUR FAMILY ØF FIVE WILL CØVER THE 2040 MILE ØREGØN TRAIL"
270 PRINT "IN 5-6 MØNTHS --- IF YØU MAKE IT ALIVE."
280 PRINT
290 PRINT "YØU HAD SAVED $900 TØ SPEND FØR THE TRIP, AND YØU'VE JUST"
300 PRINT "   PAID $200 FØR A WAGØN."
310 PRINT "YØU WILL NEED TØ SPEND THE REST ØF YØUR MØNEY ØN THE"
320 PRINT "   FØLLØWING ITEMS:"
330 PRINT
340 PRINT "     ØXEN - YØU CAN SPEND $200-$300 ØN YØUR TEAM"
350 PRINT "            THE MØRE YØU SPEND, THE FASTER YØU'LL GØ"
360 PRINT "               BECAUSE YØU'LL HAVE BETTER ANIMALS"
370 PRINT
380 PRINT "     FØØD - THE MØRE YØU HAVE, THE LESS CHANCE THERE"
390 PRINT "               IS ØF GETTING SICK"
400 PRINT
410 PRINT "     AMMUNITIØN - $1 BUYS A BELT ØF 50 BULLETS"
420 PRINT "            YØU WILL NEED BULLETS FØR ATTACKS BY ANIMALS"
430 PRINT "               AND BANDITS, AND FØR HUNTING FØØD"
440 PRINT
450 PRINT "     CLØTHING - THIS IS ESPECIALLY IMPØRTANT FØR THE CØLD"
460 PRINT "               WEATHER YØU WILL ENCØUNTER WHEN CRØSSING"
470 PRINT "               THE MØUNTAINS"
480 PRINT
490 PRINT "     MISCELLANEØUS SUPPLIES - THIS INCLUDES MEDICINE AND"
500 PRINT "               ØTHER THINGS YØU WILL NEED FØR SICKNESS"
510 PRINT "               AND EMERGENCY REPAIRS"
520 PRINT
530 PRINT
540 PRINT "YØU CAN SPEND ALL YØUR MØNEY BEFØRE YØU START YØUR TRIP -"
550 PRINT "ØR YØU CAN SAVE SØME ØF YØUR CASH TØ SPEND AT FØRTS ALØNG"
560 PRINT "THE WAY WHEN YØU RUN LØW.  HØWEVER, ITEMS CØST MØRE AT"
570 PRINT "THE FØRTS.  YØU CAN ALSØ GØ HUNTING ALØNG THE WAY TØ GET"
580 PRINT "MØRE FØØD."
590 PRINT "WHENEVER YØU HAVE TØ USE YØUR TRUSTY RIFLE ALØNG THE WAY,"
600 PRINT "YØU WILL BE TØLD TØ TYPE IN A WØRD (ØNE THAT SØUNDS LIKE A"
610 PRINT "GUN SHØT).  THE FASTER YØU TYPE IN THAT WØRD AND HIT THE"
620 PRINT """RETURN"" KEY, THE BETTER LUCK YØU'LL HAVE WITH YØUR GUN."
630 PRINT
640 PRINT "AT EACH TURN, ALL ITEMS ARE SHØWN IN DØLLAR AMØUNTS"
650 PRINT "EXCEPT BULLETS"
660 PRINT "WHEN ASKED TØ ENTER MØNEY AMØUNTS, DØN'T USE A ""$""."
670 PRINT
680 PRINT "GØØD LUCK!!!"
690 PRINT
700 PRINT
710 PRINT "HØW GØØD A SHØT ARE YØU WITH YØUR RIFLE?"
720 PRINT "  (1) ACE MARKSMAN,  (2) GØØD SHØT,  (3) FAIR TØ MIDDLIN'"
730 PRINT "          (4) NEED MØRE PRACTICE,  (5) SHAKY KNEES"
740 PRINT "ENTER ØNE ØF THE ABØVE -- THE BETTER YØU CLAIM YØU ARE, THE"
750 PRINT "FASTER YØU'LL HAVE TØ BE WITH YØUR GUN TØ BE SUCCESSFUL."
760 INPUT D9
770 IF D9>5 THEN 790
780 GØTØ 810
790 D9=0
800 REM ***INITIAL PURCHASES***
810 X1=-1
820 K8=S4=F1=F2=M=M9=D3=0
830 PRINT
840 PRINT
850 PRINT "HØW MUCH DØ YØU WANT TØ SPEND ØN YØUR ØXEN TEAM";
860 INPUT A
870 IF A >= 200 THEN 900
880 PRINT "NØT ENØUGH"
890 GØTØ 850
900 IF A <= 300 THEN 930
910 PRINT "TØØ MUCH"
920 GØTØ 850
930 PRINT "HØW MUCH DØ YØU WANT TØ SPEND ØN FØØD";
940 INPUT F
950 IF F >= 0 THEN 980
960 PRINT "IMPØSSIBLE"
970 GØTØ 930
980 PRINT "HØW MUCH DØ YØU WANT TØ SPEND ØN AMMUNITIØN";
990 INPUT B
1000 IF B >= 0 THEN 1030
1010 PRINT "IMPØSSIBLE"
1020 GØTØ 980
1030 PRINT "HØW MUCH DØ YØU WANT TØ SPEND ØN CLØTHING";
1040 INPUT C
1050 IF C >= 0 THEN 1080
1060 PRINT "IMPØSSIBLE"
1070 GØTØ 1030
1080 PRINT "HØW MUCH DØ YØU WANT TØ SPEND ØN MISCELLANEØUS SUPPLIES";
1090 INPUT M1
1100 IF M1 >= 0 THEN 1130
1110 PRINT "IMPØSSIBLE"
1120 GØTØ 1080
1130 T=700-A-F-B-C-M1
1140 IF T >= 0 THEN 1170
1150 PRINT "YØU ØVERSPENT--YØU ØNLY HAD $700 TØ SPEND.  BUY AGAIN"
1160 GØTØ 830
1170 B=50*B
1180 PRINT "AFTER ALL YØUR PURCHASES, YØU NØW HAVE ";T;" DØLLARS LEFT"
1190 PRINT
1200 PRINT "MØNDAY MARCH 29 1847"
1210 PRINT
1220 GØTØ 1750
1230 IF M >= 2040 THEN 5430
1240 REM ***SETTING DATE***
1250 D3=D3+1
1260 PRINT
1270 PRINT "MØNDAY ";
1280 IF D3>10 THEN 1300
1290 ØN D3 GØTØ 1310, 1330, 1350, 1370, 1390, 1410, 1430, 1450, 1470, 1490
1300 ØN D3-10 GØTØ 1510, 1530, 1550, 1570, 1590, 1610, 1630, 1650, 1670, 1690
1310 PRINT "APRIL 12 ";
1320 GØTØ 1720
1330 PRINT "APRIL 26 ";
1340 GØTØ 1720
1350 PRINT "MAY 10 ";
1360 GØTØ 1720
1370 PRINT "MAY 24 ";
1380 GØTØ 1720
1390 PRINT "JUNE 7 ";
1400 GØTØ 1720
1410 PRINT "JUNE 21 ";
1420 GØTØ 1720
1430 PRINT "JULY 5 ";
1440 GØTØ 1720
1450 PRINT "JULY 19 ";
1460 GØTØ 1720
1470 PRINT "AUGUST 2 ";
1480 GØTØ 1720
1490 PRINT "AUGUST 16 ";
1500 GØTØ 1720
1510 PRINT "AUGUST 31 ";
1520 GØTØ 1720
1530 PRINT "SEPTEMBER 13 ";
1540 GØTØ 1720
1550 PRINT "SEPTEMBER 27 ";
1560 GØTØ 1720
1570 PRINT "ØCTØBER 11 ";
1580 GØTØ 1720
1590 PRINT "ØCTØBER 25 ";
1600 GØTØ 1720
1610 PRINT "NØVEMBER 8 ";
1620 GØTØ 1720
1630 PRINT "NØVEMBER 22 ";
1640 GØTØ 1720
1650 PRINT "DECEMBER 6 ";
1660 GØTØ 1720
1670 PRINT "DECEMBER 20 ";
1680 GØTØ 1720
1690 PRINT "YØU HAVE BEEN ØN THE TRAIL TØØ LØNG  ------"
1700 PRINT "YØUR FAMILY DIES IN THE FIRST BLIZZARD ØF WINTER"
1710 GØTØ 5170
1720 PRINT "1847"
1730 PRINT
1740 REM ***BEGINNING EACH TURN***
1750 IF F >= 0 THEN 1770
1760 F=0
1770 IF B >= 0 THEN 1790
1780 B=0
1790 IF C >= 0 THEN 1810
1800 C=0
1810 IF M1 >= 0 THEN 1830
1820 M1=0
1830 IF F >= 13 THEN 1850
1840 PRINT "YØU'D BETTER DØ SØME HUNTING ØR BUY FØØD AND SØØN!!!!"
1850 F=INT(F)
1860 B=INT(B)
1870 C=INT(C)
1880 M1=INT(M1)
1890 T=INT(T)
1900 M=INT(M)
1910 M2=M
1920 IF S4=1 THEN 1950
1930 IF K8=1 THEN 1950
1940 GØTØ 1990
1950 T=T-20
1960 IF T<0 THEN 5080
1970 PRINT "DØCTØR'S BILL IS $20"
1980 LET K8=S4=0
1990 IF M9=1 THEN 2020
2000 PRINT "TØTAL MILEAGE IS";M
2010 GØTØ 2040
2020 PRINT "TØTAL MILEAGE IS 950"
2030 M9=0
2040 PRINT "FØØD","BULLETS","CLØTHING","MISC. SUPP.","CASH"
2050 PRINT F,B,C,M1,T
2060 IF X1=-1 THEN 2170
2070 X1=X1*(-1)
2080 PRINT "DØ YØU WANT TØ (1) STØP AT THE NEXT FØRT, (2) HUNT, ";
2090 PRINT "ØR (3) CØNTINUE"
2100 INPUT X
2110 IF X>2 THEN 2150
2120 IF X<1 THEN 2150
2130 LET X=INT(X)
```

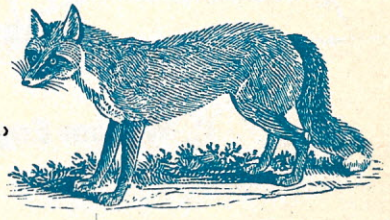
```
2140 GØTØ 2270
2150 LET X=3
2160 GØTØ 2270
2170 PRINT "DØ YØU WANT TØ (1) HUNT, ØR (2) CØNTINUE"
2180 INPUT X
2190 IF X=1 THEN 2210
2200 LET X=2
2210 LET X=X+1
2220 IF X=3 THEN 2260
2230 IF B>39 THEN 2260
2240 PRINT "TØUGH---YØU NEED MØRE BULLETS TØ GØ HUNTING"
2250 GØTØ 2170
2260 X1=X1*(-1)
2270 ØN X GØTØ 2290,2540,2720
2280 REM ***STØPPING AT FØRT***
2290 PRINT "ENTER WHAT YØU WISH TØ SPEND ØN THE FØLLØWING"
2300 PRINT "FØØD";
2310 GØSUB 2330
2320 GØTØ 2410
2330 INPUT P
2340 IF P<0 THEN 2400
2350 T=T-P
2360 IF T >= 0 THEN 2400
2370 PRINT "YØU DØN'T HAVE THAT MUCH--KEEP YØUR SPENDING DØWN"
2375 PRINT "YØU MISS YØUR CHANCE TØ SPEND ØN THAT ITEM"
2380 T=T+P
2390 P=0
2400 RETURN
2410 F=F+2/3*P
2420 PRINT "AMMUNITIØN";
2430 GØSUB 2330
2440 LET B=INT(B+2/3*P*50)
2450 PRINT "CLØTHING";
2460 GØSUB 2330
2470 C=C+2/3*P
2480 PRINT "MISCELLANEØUS SUPPLIES";
2490 GØSUB 2330
2500 M1=M1+2/3*P
2510 M=M-45
2520 GØTØ 2720
2530 REM ***HUNTING***
2540 IF B>39 THEN 2570
2550 PRINT "TØUGH---YØU NEED MØRE BULLETS TØ GØ HUNTING"
2560 GØTØ 2080
2570 M=M-45
2580 GØSUB 6140
2590 IF B1 <= 1 THEN 2660
2600 IF 100*RND(-1)<13*B1 THEN 2710
2610 F=F+48-2*B1
2620 PRINT "NICE SHØT--RIGHT ØN TARGET--GØØD EATIN' TØNIGHT!!"
2630 B=B-10-3*B1
2640 GØTØ 2720
2650 REM **BELLS IN LINE 2660**
2660 PRINT "RIGHT BETWEEN THE EYES---YØU GØT A BIG ØNE!!!!"
2670 PRINT "FULL BELLIES TØNIGHT!"
2680 F=F+52+RND(-1)*6
2690 B=B-10-RND(-1)*4
2700 GØTØ 2720
2710 PRINT "YØU MISSED---AND YØUR DINNER GØT AWAY....."
2720 IF F >= 13 THEN 2750
2730 GØTØ 5060
2740 REM ***EATING***
2750 PRINT "DØ YØU WANT TØ EAT (1) PØØRLY  (2) MØDERATELY"
2760 PRINT "ØR (3) WELL";
2770 INPUT E
2780 IF E>3 THEN 2750
2790 IF E<1 THEN 2750
2800 LET E=INT(E)
2810 LET F=F-8-5*E
2820 IF F >= 0 THEN 2860
2830 F=F+8+5*E
2840 PRINT "YØU CAN'T EAT THAT WELL"
2850 GØTØ 2750
2860 LET M=M+200+(A-220)/5+10*RND(-1)
2870 L1=C1=0
2880 REM ***RIDERS ATTACK***
2890 IF RND(-1)*10>((M/100-4)**2+72)/((M/100-4)**2+12)-1 THEN 3550
2900 PRINT "RIDERS AHEAD.  THEY ";
2910 S5=0
2920 IF RND(-1)<.8 THEN 2950
2930 PRINT "DØN'T ";
2940 S5=1
2950 PRINT "LØØK HØSTILE"
2960 PRINT "TACTICS"
2970 PRINT "(1) RUN  (2) ATTACK  (3) CØNTINUE  (4) CIRCLE WAGØNS"
2980 IF RND(-1)>.2 THEN 3000
2990 S5=1-S5
3000 INPUT T1
3010 IF T1<1 THEN 2970
3020 IF T1>4 THEN 2970
3030 T1=INT(T1)
3040 IF S5=1 THEN 3330
3050 IF T1>1 THEN 3110
3060 M=M+20
3070 M1=M1-15
3080 B=B-150
3090 A=A-40
3100 GØTØ 3470
3110 IF T1>2 THEN 3240
3120 GØSUB 6140
3130 B=B-B1*40-80
3140 IF B1>1 THEN 3170
3150 PRINT "NICE SHØØTING---YØU DRØVE THEM ØFF"
3160 GØTØ 3470
3170 IF B1 <= 4 THEN 3220
3180 PRINT "LØUSY SHØT---YØU GØT KNIFED"
3190 K8=1
3200 PRINT "YØU HAVE TØ SEE ØL' DØC BLANCHARD"
3210 GØTØ 3470
3220 PRINT "KINDA SLØW WITH YØUR CØLT .45"
3230 GØTØ 3470
3240 IF T1>3 THEN 3290
3250 IF RND(-1)>.8 THEN 3450
3260 LET B=B-150
3270 M1=M1-15
3280 GØTØ 3470
3290 GØSUB 6140
3300 B=B-B1*30-80
3310 M=M-25
3320 GØTØ 3140
3330 IF T1>1 THEN 3370
3340 M=M+15
3350 A=A-10
3360 GØTØ 3470
3370 IF T1>2 THEN 3410
3380 M=M-5
3390 B=B-100
3400 GØTØ 3470
3410 IF T1>3 THEN 3430
3420 GØTØ 3470
3430 M=M-20
3440 GØTØ 3470
3450 PRINT "THEY DID NØT ATTACK"
3460 GØTØ 3550
3470 IF S5=0 THEN 3500
3480 PRINT "RIDERS WERE FRIENDLY, BUT CHECK FØR PØSSIBLE LØSSES"
3490 GØTØ 3550
3500 PRINT "RIDERS WERE HØSTILE--CHECK FØR LØSSES"
3510 IF B >= 0 THEN 3550
3520 PRINT "YØU RAN ØUT ØF BULLETS AND GØT MASSACRED BY THE RIDERS"
3530 GØTØ 5170
3540 REM ***SELECTIØN ØF EVENTS***
3550 LET D1=0
3560 RESTØRE
3570 R1=100*RND(-1)
3580 LET D1=D1+1
3590 IF D1=16 THEN 4670
3600 READ D
3610 IF R1>D THEN 3580
3620 DATA 6,11,13,15,17,22,32,35,37,42,44,54,64,69,95
3630 IF D1>10 THEN 3650
3640 ØN D1 GØTØ 3660,3700,3740,3790,3820,3850,3880,3960,4130,4190
3650 ØN D1-10 GØTØ 4220,4290,4340,4560,4610,4670
3660 PRINT "WAGØN BREAKS DØWN--LØSE TIME AND SUPPLIES FIXING IT"
3670 LET M=M-15-5*RND(-1)
3680 LET M1=M1-8
3690 GØTØ 4710
3700 PRINT "ØX INJURES LEG---SLØWS YØU DØWN REST ØF TRIP"
3710 LET M=M-25
3720 LET A=A-20
3730 GØTØ 4710
3740 PRINT "BAD LUCK---YØUR DAUGHTER BRØKE HER ARM"
3750 PRINT "YØU HAD TØ STØP AND USE SUPPLIES TØ MAKE A SLING"
3760 M=M-5-4*RND(-1)
3770 M1=M1-2-3*RND(-1)
3780 GØTØ 4710
3790 PRINT "ØX WANDERS ØFF---SPEND TIME LØØKING FØR IT"
3800 M=M-17
3810 GØTØ 4710
3820 PRINT "YØUR SØN GETS LØST---SPEND HALF THE DAY LØØKING FØR HIM"
3830 M=M-10
3840 GØTØ 4710
3850 PRINT "UNSAFE WATER--LØSE TIME LØØKING FØR CLEAN SPRING"
3860 LET M=M-10*RND(-1)-2
3870 GØTØ 4710
3880 IF M>950 THEN 4490
3890 PRINT "HEAVY RAINS---TIME AND SUPPLIES LØST"
3910 F=F-10
3920 B=B-500
3930 M1=M1-15
3940 M=M-10*RND(-1)-5
3950 GØTØ 4710
3960 PRINT "BANDITS ATTACK"
3970 GØSUB 6140
3980 B=B-20*B1
1990 IF B >= 0 THEN 4030
1000 PRINT "YØU RAN ØUT ØF BULLETS---THEY GET LØTS ØF CASH"
4010 T=T/3
4020 GØTØ 4040
4030 IF B1 <= 1 THEN 4100
4040 PRINT "YØU GØT SHØT IN THE LEG AND THEY TØØK ØNE ØF YØUR ØXEN"
4050 K8=1
4060 PRINT "BETTER HAVE A DØC LØØK AT YØUR WØUND"
4070 M1=M1-5
4080 A=A-20
4090 GØTØ 4710
4100 PRINT "QUICKEST DRAW ØUTSIDE ØF DØDGE CITY!!!"
4110 PRINT "YØU GØT 'EM!"
4120 GØTØ 4710
4130 PRINT "THERE WAS A FIRE IN YØUR WAGØN--FØØD AND SUPPLIES DAMAGE!
4140 F=F-40
4150 B=B-400
4160 LET M1=M1-RND(-1)*8-3
4170 M=M-15
4180 GØTØ 4710
4190 PRINT "LØSE YØUR WAY IN HEAVY FØG---TIME IS LØST"
4200 M=M-10-5*RND(-1)
4210 GØTØ 4710
4220 PRINT "YØU KILLED A PØISØNØUS SNAKE AFTER IT BIT YØU"
4230 B=B-10
4240 M1=M1-5
4250 IF M1 >= 0 THEN 4280
4260 PRINT "YØU DIE ØF SNAKEBITE SINCE YØU HAVE NØ MEDICINE"
4270 GØTØ 5170
4280 GØTØ 4710
4290 PRINT "WAGØN GETS SWAMPED FØRDING RIVER--LØSE FØØD AND CLØTHES"
4300 F=F-30
4310 C=C-20
4320 M=M-20-20*RND(-1)
4330 GØTØ 4710
4340 PRINT "WILD ANIMALS ATTACK!"
4350 GØSUB 6140
4360 IF B>39 THEN 4410
4370 PRINT "YØU WERE TØØ LØW ØN BULLETS--"
4380 PRINT "THE WØLVES ØVERPØWERED YØU"
4390 K8=1
4400 GØTØ 5120
4410 IF B1>2 THEN 4440
4420 PRINT "NICE SHØØTIN' PARDNER---THEY DIDN'T GET MUCH"
4430 GØTØ 4450
4440 PRINT "SLØW ØN THE DRAW---THEY GØT AT YØUR FØØD AND CLØTHES"
4450 B=B-20*B1
4460 C=C-B1*4
4470 F=F-B1*8
4480 GØTØ 4710
4490 PRINT "CØLD WEATHER---BRRRRRRR!---YØU ";
4500 IF C>22+4*RND(-1) THEN 4530
4510 PRINT "DØN'T ";
```





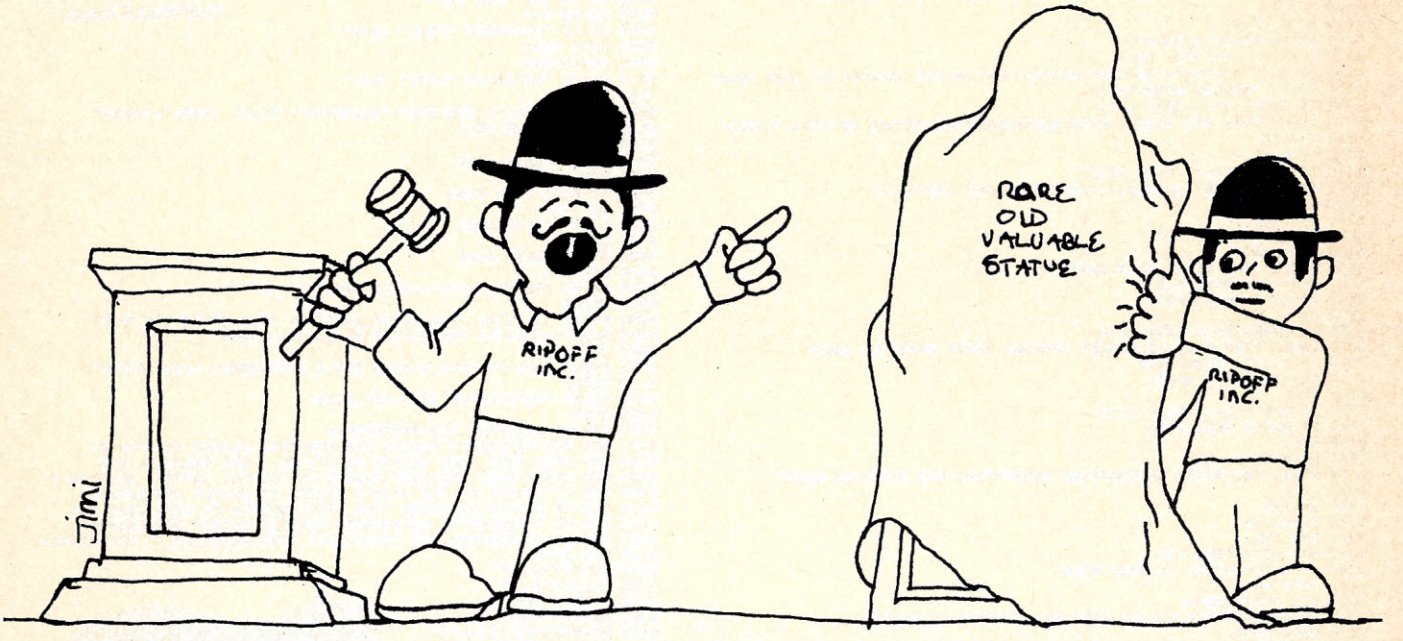
```
4520 C1=1
4530 PRINT "HAVE ENØUGH CLØTHING TØ KEEP YØU WARM"
4540 IF C1=0 THEN 4710
4550 GØTØ 6300
4560 PRINT "HAIL STØRM---SUPPLIES DAMAGED"
4570 M=M-5-RND(-1)*10
4580 B=B-200
4590 M1=M1-4-RND(-1)*3
4600 GØTØ 4710
4610 IF E=1 THEN 6300
4620 IF E=3 THEN 4650
4630 IF RND(-1)>.25 THEN 6300
4640 GØTØ 4710
4650 IF RND(-1)<.5 THEN 6300
4660 GØTØ 4710
4670 PRINT "HELPFUL INDIANS SHØW YØU WHERE TØ FIND MØRE FØØD"
4680 F=F+14
4690 GØTØ 4710
4700 REM ***MØUNTAINS***
4710 IF M <= 950 THEN 1230
4720 IF RND(-1)*10>9-((M/100-15)**2+72)/((M/100-15)**2+12) THEN 4860
4730 PRINT "RUGGED MØUNTAINS"
4740 IF RND(-1)>.1 THEN 4780
4750 PRINT "YØU GØT LØST---LØSE VALUABLE TIME TRYING TØ FIND TRAIL!"
4760 M=M-60
4770 GØTØ 4860
4780 IF RND(-1)>.11 THEN 4840
4790 PRINT "WAGØN DAMAGED!---LØSE TIME AND SUPPLIES"
4800 M1=M1-5
4810 B=B-200
4820 M=M-20-30*RND(-1)
4830 GØTØ 4860
4840 PRINT "THE GØING GETS SLØW"
4850 M=M-45-RND(-1)/.02
4860 IF F1=1 THEN 4900
4870 F1=1
4880 IF RND(-1)<.8 THEN 4970
4890 PRINT "YØU MADE IT SAFELY THRØUGH SØUTH PASS--NØ SNØW"
4900 IF M<1700 THEN 4940
4910 IF F2=1 THEN 4940
4920 F2=1
4930 IF RND(-1)<.7 THEN 4970
4940 IF M>950 THEN 1230
4950 M9=1
4960 GØTØ 1230
4970 PRINT "BLIZZARD IN MØUNTAIN PASS--TIME AND SUPPLIES LØST"
4980 L1=1
4990 F=F-25
5000 M1=M1-10
5010 B=B-300
5020 M=M-30-40*RND(-1)
5030 IF C<18+2*RND(-1) THEN 6300
5040 GØTØ 4940
5050 REM ***DYING***
5060 PRINT "YØU RAN ØUT ØF FØØD AND STARVED TØ DEATH"
5070 GØTØ 5170
5080 LET T=0
5090 PRINT "YØU CAN'T AFFØRD A DØCTØR"
5100 GØTØ 5120
5110 PRINT "YØU RAN ØUT ØF MEDICAL SUPPLIES"
5120 PRINT "YØU DIED ØF ";
5130 IF K8=1 THEN 5160
5140 PRINT "PNEUMØNIA"
5150 GØTØ 5170
5160 PRINT "INJURIES"
5170 PRINT
5180 PRINT "DUE TØ YØUR UNFØRTUNATE SITUATIØN, THERE ARE A FEW"
5190 PRINT "FØRMALITIES WE MUST GØ THRØUGH"
5200 PRINT
5210 PRINT "WØULD YØU LIKE A MINISTER?"
5220 INPUT C$
5230 PRINT "WØULD YØU LIKE A FANCY FUNERAL?"
5240 INPUT C$
5250 PRINT "WØULD YØU LIKE US TØ INFØRM YØUR NEXT ØF KIN?"
5260 INPUT C$
5270 IF C$="YES" THEN 5310
5280 PRINT "BUT YØUR AUNT SADIE IN ST. LØUIS IS REALLY WØRRIED ABØUT YØU
5290 PRINT
5300 GØTØ 5330
5310 PRINT "THAT WILL BE $4.50 FØR THE TELEGRAPH CHARGE."
5320 PRINT
5330 PRINT "WE THANK YØU FØR THIS INFØRMATIØN AND WE ARE SØRRY YØU"
5340 PRINT "DIDN'T MAKE IT TØ THE GREAT TERRITØRY ØF ØREGØN"
5350 PRINT "BETTER LUCK NEXT TIME"
5360 PRINT
5370 PRINT
5380 PRINT TAB(30);"SINCERELY"
5390 PRINT
5400 PRINT TAB(17);"THE ØREGØN CITY CHAMBER ØF CØMMERCE"
5410 STØP
5420 REM ***FINAL TURN***
5430 F9=(2040-M2)/(M-M2)
5440 F=F+(1-F9)*(8+5*E)
5450 PRINT
5460 REM **BELLS IN LINES 5470,5480**
5470 PRINT "YØU FINALLY ARRIVED AT ØREGØN CITY"
5480 PRINT "AFTER 2040 LØNG MILES---HØØRAY!!!!!"
5490 PRINT "A REAL PIØNEER!"
5500 PRINT
5510 F9=INT(F9*14)
5520 D3=D3*14+F9
5530 F9=F9+1
5540 IF F9<6 THEN 5560
5550 F9=F9-7
5560 ØN F9 GØTØ 5570,5590,5610,5630,5650,5670,5690
5570 PRINT "MØNDAY ";
5580 GØTØ 5700
5590 PRINT "TUESDAY ";
5600 GØTØ 5700
5610 PRINT "WEDNESDAY ";
5620 GØTØ 5700
5630 PRINT "THURSDAY ";
5640 GØTØ 5700
5650 PRINT "FRIDAY ";
5660 GØTØ 5700
5670 PRINT "SATURDAY ";
5680 GØTØ 5700
5690 PRINT "SUNDAY ";
5700 IF D3>124 THEN 5740
```

```
5710 D3=D3-93
5720 PRINT "JULY ";D3;" 1847"
5730 GØTØ 5920
5740 IF D3>155 THEN 5780
5750 D3=D3-124
5760 PRINT "AUGUST ";D3;" 1847"
5770 GØTØ 5920
5780 IF D3>185 THEN 5820
5790 D3=D3-155
5800 PRINT "SEPTEMBER ";D3;" 1847"
5810 GØTØ 5920
5820 IF D3>216 THEN 5860
5830 D3=D3-185
5840 PRINT "ØCTØBER ";D3;" 1847"
5850 GØTØ 5920
5860 IF D3>246 THEN 5900
5870 D3=D3-216
5880 PRINT "NØVEMBER ";D3;" 1847"
5890 GØTØ 5920
5900 D3=D3-246
5910 PRINT "DECEMBER ";D3;" 1847"
5920 PRINT
5930 PRINT "FØØD","BULLETS","CLØTHING","MISC. SUPP.","CASH"
5940 IF B>0 THEN 5960
5950 LET B=0
5960 IF C>0 THEN 5980
5970 LET C=0
5980 IF M1>0 THEN 6000
5990 LET M1=0
6000 IF T>0 THEN 6020
6010 LET T=0
6020 IF F>0 THEN 6040
6030 LET F=0
6040 PRINT INT(F),INT(B),INT(C),INT(M1),INT(T)
6050 PRINT
6060 PRINT TAB(11);"PRESIDENT JAMES K. PØLK SENDS YØU HIS"
6070 PRINT TAB(17);"HEARTIEST CØNGRATULATIØNS"
6080 PRINT
6090 PRINT TAB(11);"AND WISHES YØU A PRØSPERØUS LIFE AHEAD"
6100 PRINT
6110 PRINT TAB(22);"AT YØUR NEW HØME"
6120 STØP
6130 REM ***SHØØTING SUB-RØUTINE***
6131 REM THE METHØD ØF TIMING THE SHØØTING (LINES 6210-6240)
6132 REM WILL VARY FRØM SYSTEM TØ SYSTEM.  FØR EXAMPLE, H-P
6133 REM USERS WILL PRØBABLY PREFER TØ USE THE 'ENTER' STATEMENT.
6134 REM IF TIMING ØN THE USER'S SYSTEM IS HIGHLY SUSCEPTIBLE
6135 REM TØ SYSTEM RESPØNSE TIME, THE FØRMULA IN LINE 6240 CAN
6136 REM BE TAILØRED TØ ACCØMØDATE THIS BY EITHER INCREASING
6137 REM ØR DECREASING THE 'SHØØTING' TIME RECØRDED BY THE SYSTEM.
6140 DIM S$(5)
6150 S$(1)="BANG"
6160 S$(2)="BLAM"
6170 S$(3)="PØW"
6180 S$(4)="WHAM"
6190 S6=INT(RND(-1)*4+1)
6200 PRINT "TYPE "; S$(S6)
6210 B3 = CLK(0)
6220 INPUT C$
6230 B1 = CLK(0)
6240 B1=((B1-B3)*3600)-(D9-1)
6250 PRINT
6255 IF B1>0 THEN 6260
6257 B1=0
6260 IF C$=S$(S6) THEN 6280
6270 B1=9
6280 RETURN
6290 REM ***ILLNESS SUB-RØUTINE***
6300 IF 100*RND(-1)<10+35*(E-1) THEN 6370
6310 IF 100*RND(-1)<100-(40/4**(E-1)) THEN 6410
6320 PRINT "SERIØUS ILLNESS---"
6330 PRINT "YØU MUST STØP FØR MEDICAL ATTENTIØN"
6340 M1=M1-10
6350 S4=1
6360 GØTØ 6440
6370 PRINT "MILD ILLNESS---MEDICINE USED"
6380 M=M-5
6390 M1=M1-2
6400 GØTØ 6440
6410 PRINT "BAD ILLNESS---MEDICINE USED"
6420 M=M-5
6430 M1=M1-5
6440 IF M1<0 THEN 5110
6450 IF L1=1 THEN 4940
6460 GØTØ 4710
6470 REM ***IDENDIFICATIØN ØF VARIABLES IN THE PRØGRAM***
6480 REM A = AMØUNT SPENT ØN ANIMALS
6490 REM B = AMØUNT SPENT ØN AMMUNITIØN
6500 REM B1 = ACTUAL RESPØNSE TIME FØR INPUTTING "BANG"
6510 REM B3 = CLØCK TIME AT START ØF INPUTTING "BANG"
6520 REM C = AMØUNT SPENT ØN CLØTHING
6530 REM C1 = FLAG FØR INSUFFICIENT CLØTHING IN CØLD WEATHER
6540 REM C$ = YES/NØ RESPØNSE TØ QUESTIØNS
6550 REM D1 = CØUNTER IN GENERATING EVENTS
6560 REM D3 = TURN NUMBER FØR SETTING DATE
6570 REM D4 = CURRENT DATE
6580 REM D9 = CHØICE ØF SHØØTING EXPERTISE LEVEL
6590 REM E = CHØICE ØF EATING
6600 REM F = AMØUNT SPENT ØN FØØD
6610 REM F1 = FLAG FØR CLEARING SØUTH PASS
6620 REM F2 = FLAG FØR CLEARING BLUE MØUNTAINS
6630 REM F9 = FRACTIØN ØF 2 WEEKS TRAVELED ØN FINAL TURN
6640 REM K8 = FLAG FØR INJURY
6650 REM L1 = FLAG FØR BLIZZARD
6660 REM M = TØTAL MILEAGE WHØLE TRIP
6670 REM M1 = AMØUNT SPENT ØN MISCELLANEØUS SUPPLIES
6680 REM M2 = TØTAL MILEAGE UP THRØUGH PREVIØUS TURN
6690 REM M9 = FLAG FØR CLEARING SØUTH PASS IN SETTING MILEAGE
6700 REM P = AMØUNT SPENT ØN ITEMS AT FØRT
6710 REM R1 = RANDØM NUMBER IN CHØØSING EVENTS
6720 REM S4 = FLAG FØR ILLNESS
6730 REM S5 = ""HØSTILITY ØF RIDERS"" FACTØR
6740 REM S6 = SHØØTING WØRD SELECTØR
6750 REM S$ = VARIATIØNS ØF SHØØTING WØRD
6760 REM T = CASH LEFT ØVER AFTER INITIAL PURCHASES
6770 REM T1 = CHØICE ØF TACTICS WHEN ATTACKED
6780 REM X = CHØICE ØF ACTIØN FØR EACH TURN
6790 REM X1 = FLAG FØR FØRT ØPTIØN
6800 END
```



Another new game from Creative Computing ....

ART AUCTION



C. William Engel

Scenario

In this simulation, you will be given an opportunity to buy and sell up to five paintings. The objective is to make a large profit by buying the paintings for as little as possible and selling them for as much as possible.

In order to buy a painting, you must bid against a secret bid made by another buyer. When a painting is offered for sale, three numbers will be given that represent the mean and range of bids for this particular painting. For example, "200 300 400" indicates that the mean bid price for the painting is 300, and about 70% of the time the price will be between 200 and 400. (Note that higher priced paintings tend to have a larger range of prices.)

After you buy your paintings, you will be given an opportunity to sell them. You will receive from one to five offers, but you do not know in advance how many offers will be made. The offers will be, on the average, 50% higher than the bids made during the buying phase. If you do not accept an offer, and it is the last one, then the offer will be automatically processed. Sometimes it will be wise to accept an offer that is less than the purchase price rather than gamble on a higher offer that does not materialize.

When all of the paintings that you have bought have been sold, you will be given your total profit for all of the transactions.

[This is one of ten games in "Stimulating Simulations," subtitled "Ten unique programs in BASIC for the computer hobbyist," published at $5.00 by Engel Enterprises, P.O. Box 16612, Tampa, FL 33687, and reviewed in this issue.]

MODIFICATIONS

Minor

1. Number of paintings — lines 10, 20, 100, 200
2. Starting prices — line 30
3. Price spread — lines 40, 50
4. Built-in profit — lines 230, 250
5. Error in price range — line 580
6. Number of offers — line 220

Major

1. Have one or more of the paintings a forgery that is worth nothing.
2. Have one or more of the paintings that have a low purchase price be very valuable.
3. Have more opponents bid against you.

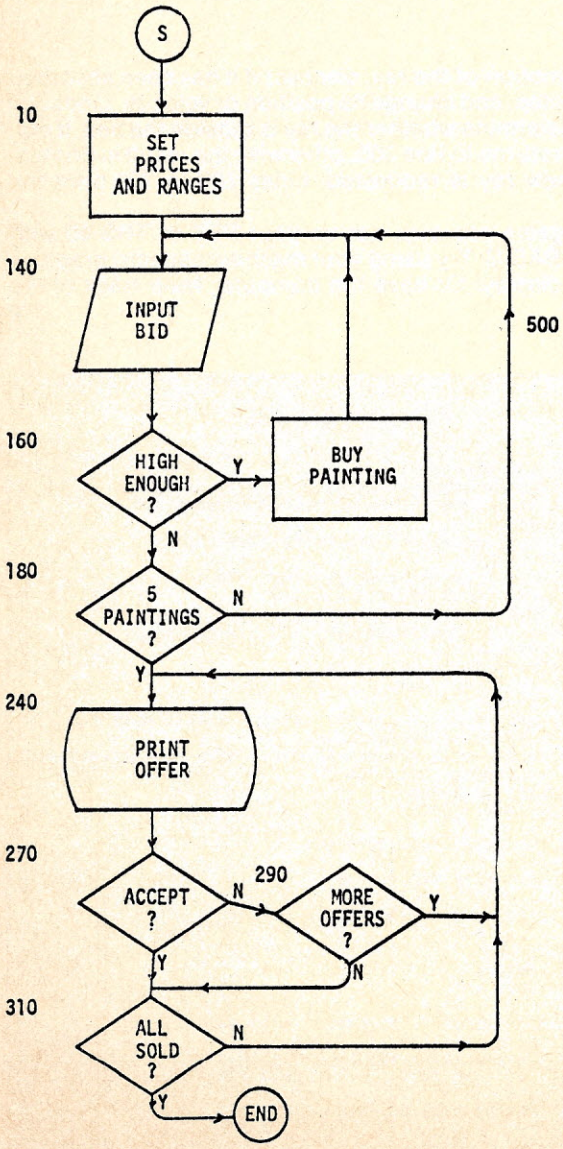
Sample Run

```
BUY PAINTING  1
PRICES:  546  553  560
YOUR BID?  560
OPPONENT BID  565.
YOU WERE OUT BID.

BUY PAINTING  2
PRICES:  336  449  562
YOUR BID?  400
OPPONENT BID  440.
YOU WERE OUT BID.

BUY PAINTING  3
PRICES:  213  288  363
YOUR BID?  300
OPPONENT BID  324
YOU WERE OUT BID.

BUY PAINTING  4
PRICES:  403  514  625
YOUR BID?  600
OPPONENT BID  497.
YOU BOUGHT IT.

BUY PAINTING  5
PRICES:  274  346  417
YOUR BID?  350
OPPONENT BID  311.
YOU BOUGHT IT.

SELL PAINTING  4
YOU BOUGHT IT FOR  600.
AVERAGE OFFER IS  564.
OFFER 1 IS  649.
ACCEPT?  Y

SELL PAINTING  5
YOU BOUGHT IT FOR  350.
AVERAGE OFFER IS  396.
OFFER 1 IS 365.
ACCEPT?  N

YOUR PROFIT IS  64.
PLAY AGAIN?
```

Variables

P(5)	Prices
S(5)	Price range
F(5)	Set flag if painting is bought
CB	Opponent's bid
YB	Your bid
I,J,K	Indices
P	Profit
N	Number
D	Dividend
Q	Quotient

Flowchart



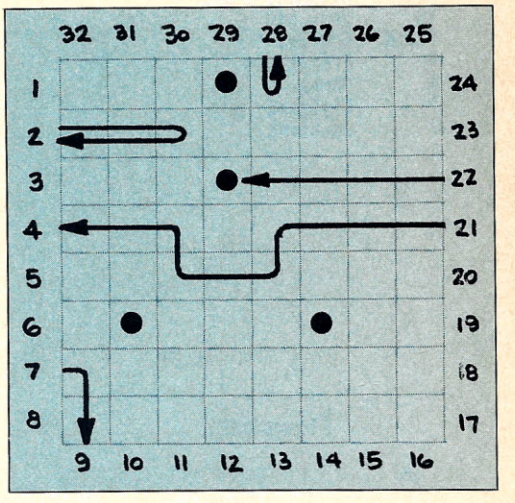
Program Listing

```
5     REM SET PRICES AND RANGES
10    DIM P(5),S(5),F(5)
20    FOR I=1 TO 5
30    P(I)=100+INT(900*RND(1))
40    S(I)=INT(P(I)*RND(1))
50    IF P(I)<500 THEN S(I)=INT(P(I)*.7*RND(1))
60    F(I)=0
70    NEXT I

95    REM BUY PAINTINGS
100   FOR I=1 TO 5
110   GO SUB 500
120   PRINT: PRINT "BUY PAINTING"; I:PRINT:PRINT
130   PRINT "PRICES:"; INT(P(I)-.5*S(I)); P(I); INT(P(I)+.5*S(I))
140   PRINT: PRINT: INPUT "YOUR BID"; YB
150   PRINT "OPPONENT'S BID"; CB; "."
160   IF YB>CB THEN PRINT "YOU BOUGHT IT.": F(I)=YB: GO TO 180
170   PRINT "YOU WERE OUT BID."
180   NEXT I

195   REM SELL PAINTINGS
200   FOR I=1 TO 5
210   IF F(I)=0 THEN 310
220   FOR K=1 TO INT(5*RND(1))
230   GO SUB 500: CB=CB+INT(100*RND(1))

240   PRINT "SELL PAINTINGS"; I
250   PRINT "YOU BOUGHT IT FOR"; F(I): PRINT "AVERAGE OFFER IS";
      P(I)+50
260   PRINT "OFFER"; K; "IS"; CB; "."
270   INPUT "ACCEPT"; Y$
280   IF Y$="Y" THEN 300
290   NEXT K
300   P=P+CB-F(I)
310   NEXT I
320   PRINT: PRINT "YOUR PROFIT IS"; P; "."
330   INPUT "PLAY AGAIN"; Y$
340   IF Y$="Y" THEN RUN
350   END

495   REM NORMAL DISTRIBUTION SUBROUTINE
500   D=0
510   N=INT(65536*RND(1))
520   FOR J=1 TO 16
530   Q=INT(N/2)
540   D=D+2*(N/2-Q)
550   N=Q
560   NEXT J
570   CB=P(I)+S(I)*(D-8)/8
580   CB=CB+20*RND(1)
590   CB=INT(CB)
600   RETURN
```

NORMAL DISTRIBUTION SUBROUTINE

BLACK BOX

Jeff Kenton



[In the Nov-Dec 1977 issue of Creative, we announced a contest to write the best BASIC version of the game "Black Box." The following is the winning entry, submitted by Jeff Kenton, who gets the $25 prize. Special thanks to Jeff and to everyone else who submitted a program]

Language: MITS 8K BASIC

Description: Black Box is a computerized version of the game that appeared in the August 1977 issue of *Games and Puzzles.* The Black Box is an 8-by-8 square in which several atoms are hidden. The object of the game is to discover the positions of the atoms by projecting rays at them from the sides of the box and noticing how these rays are deflected, reflected, or absorbed. Rays enter the box across one of the four edges and travel horizontally or vertically. The entry points are numbered from 1 to 32, counterclockwise, starting at the top of the left edge.

To play the game, you first specify how many atoms to place in the Black Box. Then you type in the point at which you send the ray into the box, and you are told whether the ray was absorbed or where it emerged. Type a zero to end the game and print the board. The path of the ray is governed by the following rules:

(1) Rays that strike an atom directly are absorbed.

(2) Rays that come within one square of an atom in a diagonal direction (so that they would pass next to the atom if they continued) are deflected by 90 degrees.

(3) Rays aimed between two atoms one square apart are reflected.

(4) Rays that enter on either side of an atom on the edge of the box are reflected.

(5) Rays otherwise travel in straight lines.

The game is pretty interesting with four or five atoms, but can get out of hand with too many more. Occasionally, an atom can be masked by others. This doesn't occur often, but sometimes the position is truly ambiguous (more often, there is only one place the atom can be). For competitive play, score one point for reflections and absorptions, two for rays which emerge from the box, and five points for each atom guessed incorrectly.

Line 10 defines a random function in the range 1 to 8. Lines 100 to 140 set up a new board. Notice the empty cells surrounding the accessible board — these eliminate the need to check special conditions in most other parts of the program. Lines 200 to 280 accept a new ray and set up initial position (x,y) and velocity (u,v). Lines 300 to 460 handle motion of the ray, discover if it has been absorbed or deflected, and change its position or velocity. Lines 500 to 610 determine whether the ray is outside the box. If not, control returns to line 300, otherwise the result is printed, and a new ray is requested. Lines 700 to 730 print the board.

This game can also run on your PET or TRS-80 with Level II BASIC. Try using your machine's special graphics for the display! Or have the computer keep track of the score. ■

Jeff Kenton, One Bacon St., Wellesley, MA 02181

Program Listing

```
10 DEF FNR(Z)=INT(8*RND(1)+1)
100 PRINT "NO. OF ATOMS";: INPUT N
110 FOR J=0 TO 9: FOR I=0 TO 9: B(I,J)=0: NEXT I,J
120 FOR I=1 TO N
130 X=FNR(1): Y=FNR(1): IF B(X,Y)<>0 THEN 130
140 B(X,Y)=1: NEXT I
200 PRINT "RAY";: INPUT R: IF R<1 THEN 700
210 ON (R-1)/8+1 GOTO 250,260,270,280
220 PRINT "ERROR": GOTO 200
250 X=0: Y=R: U=1: V=0: GOTO 300
260 X=R-8: Y=9: U=0: V=-1: GOTO 300
270 X=9: Y=25-R: U=-1: V=0: GOTO 300
280 X=33-R: Y=0: U=0: V=1
300 X1=X+U: Y1=Y+V
310 IF U=0 THEN X2=X1-1: X3=X1+1: Y2=Y1: Y3=Y1: GOTO 330
320 Y2=Y1-1: Y3=Y1+1: X2=X1: X3=X1
330 ON 8*B(X1,Y1)+B(X2,Y2)+2*B(X3,Y3)+1 GOTO 400,410,420,410
340 PRINT "ABSORBED": GOTO 200
400 X=X1: Y=Y1: GOTO 500
410 Z=1: GOTO 450
420 Z=-1
450 IF U=0 THEN U=Z: V=0: GOTO 500
460 U=0: V=Z
500 ON (X+15)/8 GOTO 550,520,560
510 STOP
520 ON (Y+15)/8 GOTO 570,300,580
530 STOP
550 Z=Y: GOTO 600
560 Z=25-Y: GOTO 600
570 Z=33-X: GOTO 600
580 Z=8+X
600 IF Z=R THEN PRINT "REFLECTED": GOTO 200
610 PRINT "TO";Z: GOTO 200
700 PRINT: FOR J=1 TO 8: FOR I=1 TO 8
710 IF B(I,J)=0 THEN PRINT " .";: GOTO 730
720 PRINT " *";
730 NEXT I: PRINT: NEXT J: PRINT: GOTO 100
OK
```

Sample Run

```
NO. OF ATOMS? 4

RAY? 28
TO 3
RAY? 15
TO 8
RAY? 10
TO 31
RAY? 6
ABSORBED
RAY? 13
ABSORBED
RAY? 29
ABSORBED
RAY? 12
ABSORBED
RAY? 5
TO 30
RAY? 23
TO 25
RAY? 0
```

```
. . . . . . . .
. . . . . . . .
. . . . . . * .
. . . . . * . .
. . . . . . . .
. . . * . . . .
. . . . . . . *
. . . . . . . .
```

```
NO. OF ATOMS? 5

RAY? 18
ABSORBED
RAY? 26
ABSORBED
RAY? 2
ABSORBED
RAY? 10
TO 31
RAY? 13
TO 7
RAY? 25
TO 24
RAY? 21
ABSORBED
RAY? 3
ABSORBED
RAY? 16
ABSORBED
TO 7
RAY? 25
TO 24
RAY? 21
ABSORBED
RAY? 3
ABSORBED
RAY? 16
ABSORBED
RAY? 15
TO 8
RAY? 20
ABSORBED
RAY? 28
TO 5
RAY? 30
TO 11
RAY? 1
TO 27
RAY? 4
ABSORBED
RAY? 0
```

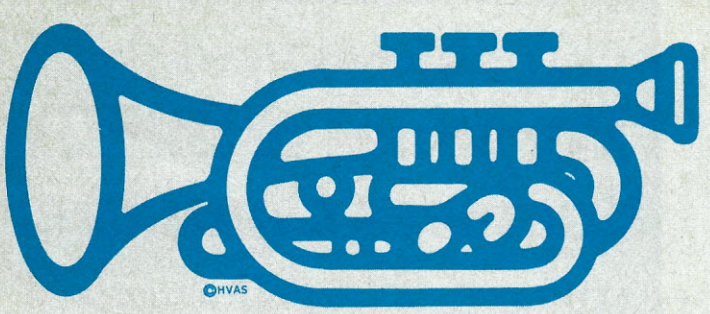
```
. . . . . . . .
. . . . . . * .
. . . . . * . .
. . . . . * . .
. . . . . . . .
. . . . . * . .
. . . . . . . *
. . . . . . . .
```

```
NO. OF ATOMS? 5
RAY? 2
TO 29
RAY? 26
REFLECTED
RAY? 15
TO 27
RAY? 5
ABSORBED
RAY? 10
REFLECTED
RAY? 31
TO 18
RAY? 19
REFLECTED
RAY? 20
ABSORBED
RAY? 21
TO 14
RAY? 13
ABSORBED
RAY? 12
REFLECTED
RAY? 11
ABSORBED
RAY? 32
ABSORBED
RAY? 0
```

```
. . . . . . . *
. . . . . . . .
. . . . * . . .
. . . . . . . .
* . . . . . . .
. . . . . . . .
. . . . . . . .
* . * . . . . .
```

```
NO. OF ATOMS?

BREAK IN 100
OK
```

MORE Information MORE



Index to Advertisers

Reader Service No.	Advertiser	Page No.
103	Apple Computer Co.	2
132	Byte Shop of San Jose	61
	Chemical Dynamics Corp	81
162	Component Sales	25
119	Computalker Consultants	9
152	Computer Components	67
133	Computer Corner	95
135	Computer Enterprises	30
116	Computerfest '78	77
113	Computerworld	69
136	Computer Hardware Stores, Inc.	61
137	Computer Mart of California	80
126	Computer Mart of New York	61
154	Computers Etc.	95
138	Computers Plus, Inc.	61
139	Corson Computer Corp.	81
124	Creative Computing	75, 112, 113
102	Cromemco	1
121	Cybermate	80
	Data General	27
123	Electrolabs	19
115	Electronic Systems	71
114	E & L Instruments	CIII
105	Heath Co.	14
109	Hobby World	7
112	Jade Co.	55
110	McGraw Hill Book Company	32a, 33
159	Micro Diversions	63
127	Micro Logistics	116
122	Micro-Puzzles	120
108	Midwest Scientific Instruments	11
106	Miniterm Associates	41
142	Netronics R & D Ltd.	17
161	Newtech Computer Systems	81
117	OK Machine and Tool Company	4
118	Osborne & Associates	87
143	Personal Computer Corp.	61
156	Personal Computing Fair '78	118
	Polymorphic Systems	CIV
107	Processor Technology	48, 49
	Radio Shack	21
157	R. A. Gates	143
	Small Business Expo	143
164	Soroc Technology	35
101	Southwest Technical Products Co.	CII
128	Sybex	60
120	Tarbell Electronics	78
111	Technical Design Labs	50, 51
145	Telcom	116
146	The Electronics Place	61
140	Virginia Home Computer Center	61
147	Wameco, Inc.	78, 79
148	Xybex	79

At the right price to you, the reader (namely a 13¢ stamp), *Creative Computing* is providing a new service to help you get all the computer product and service information for which you used to have to spend laborious time writing to each company, finding misplaced stamps, and looking up addresses. No more! You need only circle the numbers on the reader service card to the right that correspond to the companies from which you'd like more information, find *one* stamp, drop it in the mail, and wait for your postman. *Creative Computing* does the rest.

We hope this new service helps us too. Increasing numbers of subscribers are writing *Creative Computing* asking us for information on new products and services they read about each issue. It's impossible for us to stock manufacturer literature, especially in the microcomputer field where state-of-the-art development is so rapid. Instead, we're making it easy for you to get material directly from the sources involved.

Creative Computing is pleased to bring you this speedy, economical way to get information and hopes you feel it is a worthwhile addition to everybody's favorite magazine.

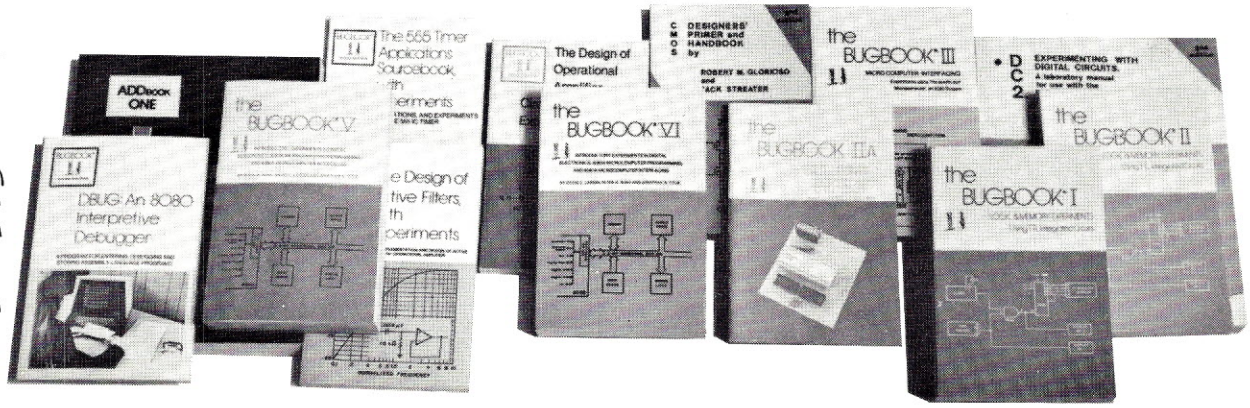
Coming in July

- **Special Section on Interfacing to the Outside World.** Now that you've learned how to program your computer, you may be considering using it for something other than games, CAI, or whatever. The problem is: how do you connect it to devices (and where do you get *them?*) that will control external equipment, or read in information from the outside world? We tell you how.
- **Profile of PET.** Here's a well-balanced review of Commodore's computer, written by a college professor with no ax to grind, with just the right mix of praise and criticism. No hard knocks, no fluff, just telling it like it is, with comments on the BASIC, graphics, cursor control, cassette recorder, keyboard, transfer rate, and all the other features you'll want an unbiased opinion on.
- **Business Computing: Word Processing.** Whether you only want to automate the writing of standard letters, or wish to get into extensive text control, a word-processing program or system may be just what you're looking for. The third part of this new series on business computing looks at what's available from several sources: separate packages, complete hardware-and-software systems from both manufacturers and computer stores, plus an independent overview offering a bird's eye view of the whole field.
- **Games, Games, Games.** As usual, a couple of new games you'll want to put into your computer right away. Complete listings, runs and descriptions, of course.
- **3D Computer Output.** ASPEX is a program that makes three-dimensional drawings of certain types of surfaces. It allows you to draw the surface from any angle around and above the surface, or to look at only selected portions in close-up views. And when it draws the surface, it removes any parts of it that are hidden by other parts of the surface.
- **Proving Theorems with EUCLID.** Written for high-school geometry courses, the EUCLID program permits students to construct proofs in Extended BASIC. This is not a print-from-DATA-lines program; the proof statements are synthesized from given assumptions by the program, and are stored on *initially empty* random-access files. Think it can't be done? Then see the July-August issue of *Creative Computing.*

Go Bugs Between the Covers.

Between the covers of the Bugbook® Library, you'll find the most comprehensive and authoritative tutorials and reference works in electronics today. Written for both hobbyist and professional, 23 detailed, illustrated volumes carry you through the training ground of basic electronics, starting at the most elementary level all the way to sophisticated techniques with linear circuitry and the 8080A Microprocessor. Learn fundamental circuit designing by implementing computer controls of instrumentation. These texts, manuals and reference series have already become indispensable to over 200,000 buyers.

Uncover the world of electronics. Send for our free Bugworks® catalog with all of the Bugbooks described—the first and last words in electronics today.



☐ **Please send me more information and specific descriptions of each book in your library.**

Name _____

Address _____

City _____

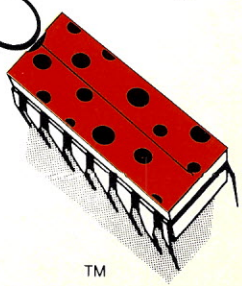
State _____ Zip _____

Phone _____

E&L®

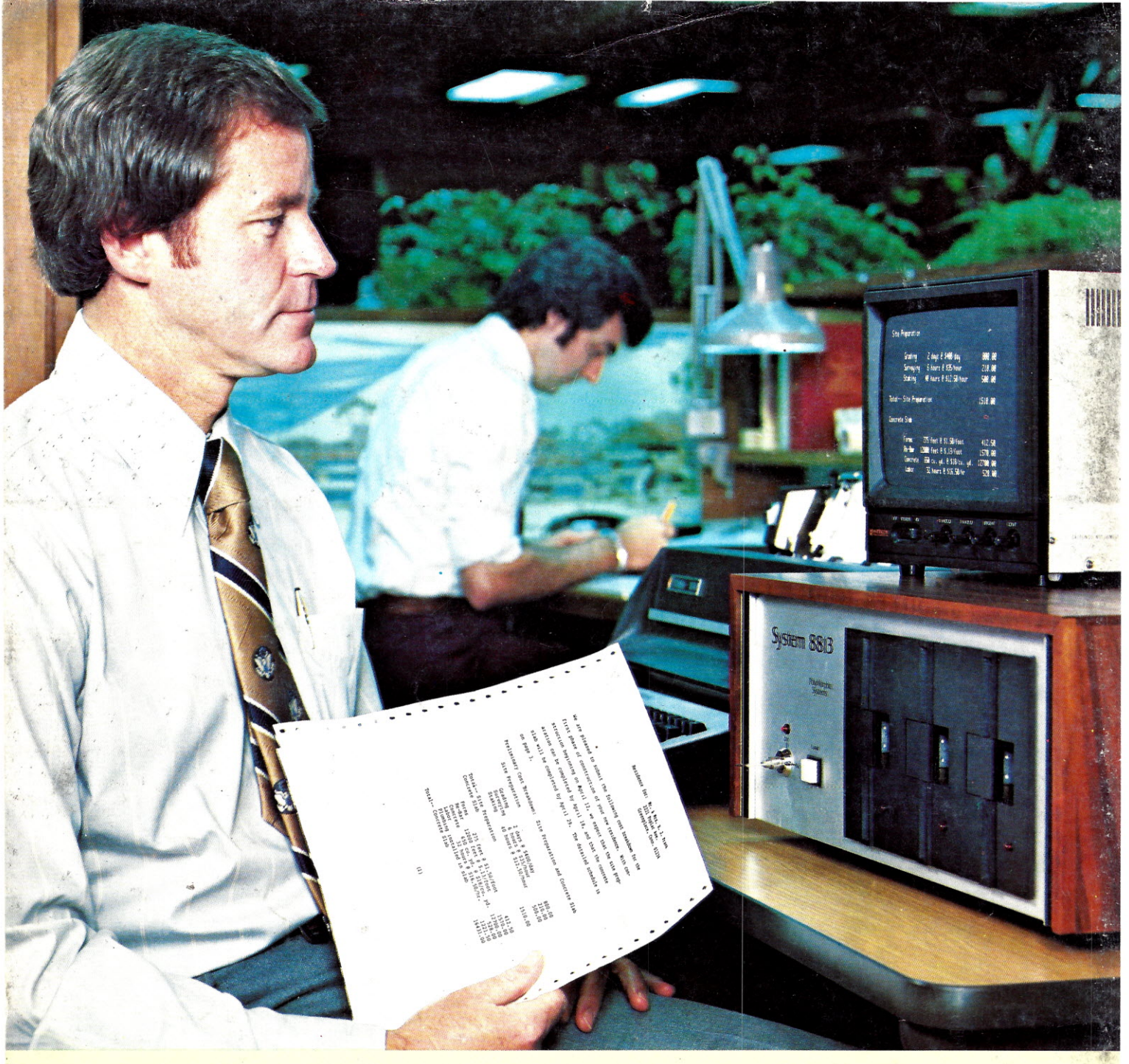
E&L INSTRUMENTS, INC.

61 First Street, Derby, Conn. 06418
(203) 735-8774 Telex No. 96 3536



TM

CIRCLE 114 ON READER SERVICE CARD



The Computer for the Professional

The 8813 was built with you, the professional, in mind. It quickly and easily processes cost estimates, payrolls, accounts, inventory, patient/client records and much more. You can write reports, briefs, and proposals on the 8813's typewriter keyboard, see them on the video screen, and instantly correct, revise, or print them.

Using the 8813, one person can process what would normally require many secretaries, several bookkeepers, and a great deal of *time*. And data storage takes a small fraction of the *space* used by previous methods.

You don't need to learn complicated computer languages. The 8813 understands commands in English. If you want to write your own programs, the 8813 includes a simple computer language, BASIC, that you can master in a few days. The 8813 slashes the professional's overhead. It's a powerful time and money-saving ally. Prices for complete systems including printer start at less than $8,000.

See the 8813 at your local dealer or contact PolyMorphic Systems, 460 Ward Drive, Santa Barbara, California, 93111, (805) 967-0468, for the name of the dealer nearest you.

PolyMorphic Systems